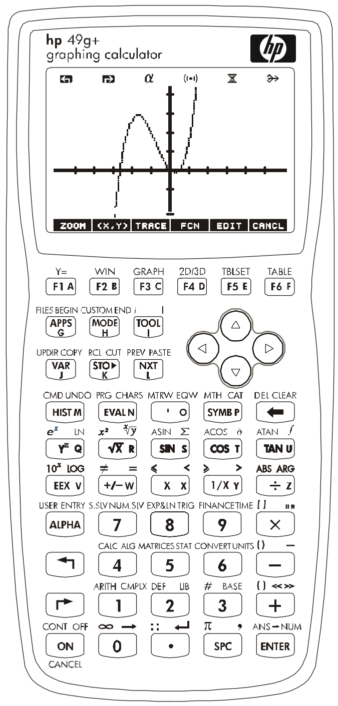
hp 49g+

grafikfähiger Taschenrechner

Benutzeranleitung



i n v e n t

ANMERKUNG

Vorwort

Sie halten nun einen kompakten, symbolischen und numerischen Rechner in Händen, der Ihnen die Berechnung und mathematische Analyse einer Vielzahl von Aufgaben in den verschiedensten Fachbereichen erleichtern wird, von elementarer Mathematik über Berechnungen im Ingenieurwesen bis hin zu anspruchsvollen wissenschaftlichen Aufgabenstellungen. Obgleich das Gerät hier aufgrund seiner kompakten Abmessungen als Taschenrechner bezeichnet wird, handelt es sich beim hp 49g+ um einen vollwertigen grafischen, programmierbaren Handheld computer.

Der hp 49g+ verfügt über zwei verschiedene Betriebsarten, einmal über den RPN-Modus (RPN=*Reverse Polish Notation - Umgekehrte Polnische Notation*) und über den ALG-Modus (ALG=*Algebra*, für weitere Details siehe die Seiten 1-11 in der Benutzeranleitung. Der RPN-Modus wurde bei Taschenrechnern eingeführt, um die Effizienz von Berechnungen zu erhöhen. In dieser Betriebsart werden zunächst die Operanden einer Berechnung (z.B., die '2' und die '3' in der Berechnung des Ergebnisses von '2+3') eingegeben und erst anschliessend der Operator (z.B., '+' bei der Berechnung von '*2+3*'). Im ALG-Modus hingegen geben Sie arithmetische Ausdrücke genauso ein, wie Sie sie auf Papier niederschreiben würden. Um die Berechnung auszuführen, benutzen wir die ENTER Taste. Diese Benutzeranleitung illustriert Beispiele für die Anwendung der verschiedenen Funktionen und Berechnungsmöglichkeiten dieses Taschenrechners in beiden Modi.

Die Kapitel dieser Benutzeranleitung sind nach Themen mit ansteigendem Schwierigkeitsgrad angeordnet. Sie beginnen mit der Einstellung der Rechner-Betriebsarten und der Anzeige-Optionen und fahren fort mit Rechenoperationen mit realen und komplexen Zahlen, Operationen mit Listen, Vektoren und Matrixen, detaillierten Beispielen grafischer Anwendungen, der Benutzung von Strings, Programmier-Grundlagen, Grafik-Programmierung, Bearbeitung von programmierten Schleifen, Kalkulus, Kalkulus-Anwendungen mit multiplen Variablen, Anwendungen mit Differentialgleichungen (einschliesslich der Laplace-Transformation, Fourier-Reihen und Fourier-Transformationen), Wahrscheinlichkeitsberechnungen und statistischen Anwendungen.

Das Herz des Rechners ist ein aufrüstbares Betriebssystem, dass Sie durch den Download neuer Versionen von der Website des Rechners aktualisieren können. Für Operationen mit Symbolen verfügt der Rechner über eine mächtiges Computer Algebra System (CAS) mit dem Sie verschiedene Betriebsarten wählen können, z.B. für komplexe Zahlen und reale Zahlen oder exakte und angenäherte Zahlen. Das Display kann so eingestellt werden, dass es Ausdrücke ähnlich wie im Lehrbuch anzeigt, was bei der Arbeit mit Matrixen, Vektoren, Brüchen, Summen, Ableitungen und Integralen nützlich sein kann. Mit der Hochgeschwindigkeits-Grafikfähigkeit des Rechners erzeugen Sie grafische Dartsllungen bequem und in kürzester Zeit.

Dank der Infrarot-Schnittstelle und des USB-Kabels, das für Ihren Rechner erhältlich ist, können Sie Ihren Rechner mit anderen Rechnern oder Computern verbinden. Durch die Infrarot- oder die USB-Hochgeschwindigkeitsverbindung können Sie schnell und effizient Daten und Programme mit anderen Rechnern oder Computern austauschen. Der Rechner verfügt über Flash-Memory-Card Anschlüsse, um die Speicherung und den Austausch von Daten mit anderen Benutzern zu ermöglichen.

Die Programmierfähigkeiten des Rechners ermöglichen Ihnen oder anderen Benutzern die Entwicklung von effizienten Anwendungen für spezielle Zwecke. Ob fortgeschrittene mathematische Anwendungen, spezifische Problemlösungen oder Datenaufzeichnung - die von Ihrem Rechner zur Verfügung gestellten Programmiersprachen machen ihn zu einem äusserst vielseitigen Gerät.

Wir hoffen, dass Ihnen ihr Rechner ein treuer, zuverlässiger Begleiter sein wird, sowohl für Ihre Anwendungen in der Ausbildung sowie im Beruf.

Inhaltsverzeichnis

Anhänge

Kapitel 1
Einführung

Dieses Kapitel ist dazu gedacht, Ihnen Grundkenntnisse zur Bedienung Ihres Rechners zu vermitteln. Die Beispiele dienen dazu, Sie mit den Grundoperationen und Einstellungen des Rechners vertraut zu machen, bevor Sie mit den eigentlichen Berechnungen beginnen.

Grundoperationen

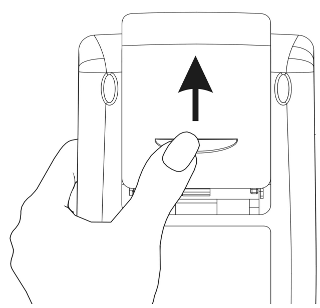
Nachfolgende Beispiele sollen Sie mit der Hardware des Rechners vertraut machen.

Batterien

Für den Rechner werden 3 AAA Batterien zur Hauptstromversorgung und eine CR2032 Lithium Batterie für das Back-Up des Datenspeichers benötigt.
Bevor Sie den Rechner in Betrieb nehmen, setzen Sie die Batterien gemäß nachfolgenden Anweisungen ein:

Installation der Batterien für die Hauptstromversorgung

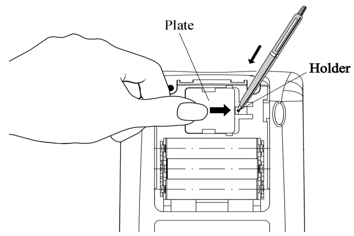
a. Öffnen Sie das Batteriefach, wie unten abgebildet.



b. Geben Sie die 3 neuen AAA Batteriren ins Hauptfach. Stellen Sie sicher, dass jede Batterie in der angegeben Richtung eingelegt wird.

Installation der Batterien für das Back-Up des Datenpeichers

a. Drücken Sie die Abdeckung nach unten. Schieben Sie den Deckel in die angegebene Richtung und heben Sie diese an.



b. Setzen Sie eine neue CR2032 Lithium Batterie ein. Stellen Sie sicher, dass die positive (+) Seite nach oben zeigt.
c. Setzen Sie den Deckel wieder auf und schieben diesen in die ursprüngliche Position.
Nachdem Sie die Batterien installiert haben, drücken Sie [ON], um den Rechner einzuschalten.
Warnung: Sobald das Symbol niedriger Batteriefüllstand angezeigt wird, müssen Sie die Batterien so schnell wie möglich austauschen. Vermeiden Sie aber beide Batterien – die für den Hautstromanschluss und für das Back-Up des Datenpeichers - gleichzeitig zu wechseln, um einen Datenverlust zu vermeiden.

Ein- und ausschalten des Rechners

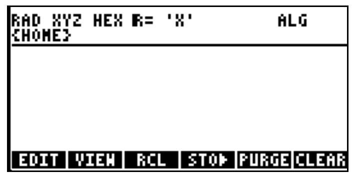
Die [ON] Taste befindet sich links unten auf der Tastatur.Drücken Sie diese Taste einmal, um Ihren Rechnern einzuschalten.Um den Rechner auszuschalten, drücken Sie die rote rechte Shift-Taste [↱] (von unten die erste Taste in der zweiten Reihe) und anschließend die Taste [ON].Beachten Sie, dass sich auf der Taste [ON] eine rote Markierung OFF in der rechten oberen Ecke, als Erinnerung an die Ausschaltfunktion des Rechners, befindet.

Kontrast des Displays einstellen

Der Kontrast des Displays kann mit den Tasten [+]und [−] bei gleichzeitig gedrückter [ON]Taste eingestellt werden. Die Tastenkombination [ON] (halten) [+] stellt das <u>Display dunkler</u>. Durch drücken der Tastenkombination [ON] (halten) [−] wird das <u>Display heller</u> eingestellt.

Anzeigen im Display des Rechners

Schalten Sie Ihren Rechner erneut ein. Das Display sollte wie folgt aussehen:



Im oberen Teil des Displays erscheinen zwei Zeilen mit den Einstellungen des Rechners. In der ersten Zeile erscheinen folgende Zeichen:

RAD XYZ HEX R= 'X'

Details über die Bedeutung dieser Angaben finden Sie in Kapitel 2.
In der zweiten Zeile erscheinen nachfolgende Zeichen: { HOME }, welche das HOME Verzeichnis als aktuelles Verzeichnis im Speicher des Rechners ausweisen. In Kapitel 2 erfahren Sie dann, dass Sie in Ihrem Rechner Daten entweder in Dateien oder Variablen speichern können. Variablen können in Verzeichnissen und Unterverzeichnissen organisiert werden. Schließlich können Sie auch eine Verzeichnisstruktur erstellen, ähnlich der wie diese in Computern verwendet wird. Sie können dann durch die Verzeichnisstruktur navigieren, um das gewünschte Verzeichnis auszuwählen. Während Sie sich durch die Verzeichnisstruktur bewegen, wird in der zweiten Zeile des Displays das entsprechende Verzeichnis zusammen mit dem Unterverzeichnis angezeigt.

Am unteren Teil des Displays befinden sich verschiedene beschriftete Schaltflächen, und zwar:

EDIT VIEW RCL STO PURGE CLEAR

welche den *Funktionstasten* F1 bis F6 zugeordnet sind:

F1 F2 F3 F4 F5 F6

Die Beschriftung der sechs Schaltflächen am unteren Rand des Rechners wechselt, abhängig vom Menü, welches gerade angezeigt wird. Aber die Funktionstaste F1 wird immer der ersten angezeigten Beschriftung zugeordnet sein, F2 der zweiten Beschriftung und so weiter.

Menüs

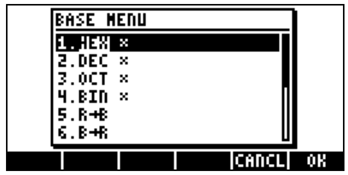
Die sechs, den Funktionstasten F1 bis F6 zugeordneten Beschriftungen, sind Teile eines Funktionsmenüs. Da der Rechner nur insgesamt sechs Funktionstasten besitzt, werden jeweils nur 6 Schaltflächen gleichzeitg angezeigt. Ein Menü kann aber auch mehr als nur sechs Einträge besitzen. Eine Gruppe von 6 Einträgen wird als Menüseite bezeichnet. Das aktuelle Menü, bekannt als das Menü TOOL (siehe unten), hat 8 Einträge, welche auf zwei Seiten angeordnet sind. Die zweite Seite, in welchem die letzten beiden Einträge enthalten sind, kann durch drücken der Taste NXT (Menü NeXT – nächstes Menü) angezeigt werden. Auf der Tastatur ist dies die dritte Taste von links in der dritten Reihe von unten. Ein weiteres drücken der Taste NXT oder drücken der Taste TOOL (dritte in der zweiten Reihe von oben), bringt Sie zurück ins Hauptmenü TOOL.

Das Menü TOOL wird im nächsten Abschnitt behandelt. An dieser Stelle möchten wir einige Menüeigenschaften zeigen, welche Ihnen bei der Benutzung Ihres Rechners hilfreich sind.

Menüs SOFT vs. CHOOSE Kästchen

Menüs oder SOFT Menüs, funktionieren in Kombination mit den sechs, am unteren Bildschirmrand befindlichen Funktionstasten (F1 bis F6). Durch drücken der entsprechenden Funktionstaste, wird die Funktion mit der entsprechenden Beschriftung aktiviert. So z.B. drücken der Taste CLEAR (F6), bei aktiviertem TOOL Menü, startet die Funktion CLEAR, welche den Inhalt der Anzeige löscht (bereinigt). Um diese Funktion auszuprobieren, geben Sie eine Zahl ein, sagen wir 1 2 3 ENTER und drücken anschließend die Taste F6 .

Die Menüs SOFT werden normalerweise zur Auswahl aus einer bestimmten Anzahl von Funktionen eingesetzt. SOFT Menüs sind aber nicht der einzige Zugriff auf Gruppen von verwandten Funktionen. Die Alternative dazu sind die sogenannten CHOOSE Kästchen. Um ein Beispiel eines CHOOSE-(Auswahl)-kästchens zu sehen, aktivieren Sie das TOOL Menü (durch drücken der Taste TOOL), anschließend verwenden Sie die Tastenkombination → BASE (der Taste 3 zugeordnet). Sie erhalten das nachfolgende CHOOSE Kästchen:



Dieses CHOOSE Kästchen ist als BASE MENU (Basismenü) beschriftet und stellte eine durchnummerierte Liste von Funktionen zur Verfügung, von 1. HEX x bis 6. B→R. Diese Anzeige stellt die erste Seite dieses CHOOSE Box Menüs dar und zeigt 6 Menüfunktionen. Mit den Pfeiltasten ▲▼, welche sich im oberen Teil der Tastatur, gleich unter den Funktionstasten F5 und F6 befinden, können Sie durch das Menü navigieren. Um eine der Funktionen zu starten, markieren Sie diese Funktion mit Hilfe der Pfeiltasten ▲▼ oder einfach durch drücken der, dem CHOOSE Kästchen zugeordneten Zahl. Nachdem Sie die Funktion ausgewählt haben, drücken Sie die Taste OK, Funktionstaste (F6). Sollten Sie z.B. die Funktion R→B (reell nach binär) auswählen wollen, könnten Sie die Tastenfolge 6 F6 drücken.

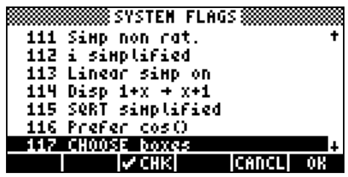
Wenn Sie direkt an den Anfang der Menü Seite in ein CHOOSE Kästchen springen möchten, geschieht dies mit der Tastenfolge ← ▲ tun. Um ans Ende der aktuellen Seite zu gelangen, benutzen Sie die Tastenfolge ← ▼. Um an den absoluten Anfang des Menüs zu gelangen, drücken Sie die Tastenfolge → ▲. Um ans absolute Ende des Menüs zu gelangen, drücken Sie die Tastenfolge → ▼.

Auswahl von SOFT Menüs oder CHOOSE Kästchen

Sie können das Format, in welchem Ihre Menüs angezeigt werden sollen, durch ändern eines System Flags im Rechner, selbst bestimmen (Ein System Flag ist eine Variable des Rechners, welche einen bestimmten Operationsmodus des Rechners überwacht. Mehr dazu erfahren Sie in Kapitel 24.) Durch umstellen des System Flags 117 können Sie die Anzeige von SOFT Menü auf CHOOSE Kästchen ändern. Um zum diesem Flag zu gelangen, benutzen Sie die Tastefolge:

MODE FLAGS ▲ ← ▲ ▼

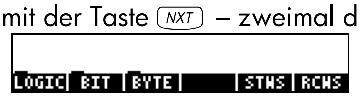
Auf Ihrem Display erscheint die folgende Anzeige, wobei die Zeile mit der Nummer 117 hervorgehoben ist:



Standardmäßig wird die Zeile wie oben aussehen. Die hervorgehobene Zeile (117 CHOOSE boxes) zeigt an, dass Ihre Anzeige im Display im Moment auf CHOOSE Kästchen steht. Ziehen Sie aber die Verwendung von Funktionstasten vor, drücken Sie CHK, Funktionstaste (F3) gefolgt von OK (F6). Drücken Sie OK (F6) ein weiteres Mal, um zur Normalanzeige des Rechners zurückzukehren.

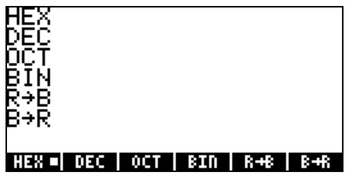
Wenn Sie die Tastenfolge → BASE, anstelle der CHOOSE Kästchen, die Sie vorhin gesehen haben, drücken, weist Ihre Anzeige sechs Einträge als Seite eins des Menüs STACK auf:

Um zwischen den einzelnen Funktionen des Menüs zu navigieren, drücken Sie die Taste NXT, um auf Seite zwei des Menüs zu gelangen oder ← PREV (der Taste NXT zugeordnet), um zur vorherigen Seite zu gelangen. Die nachfolgenden Zahlen zeigen verschiedene Seiten des Menüs BASE, auf welche mit der Taste NXT – zweimal drücken – zugegriffen werden kann.



Ein weiteres Drücken der Taste NXT, bringt uns auf die erste Menüseite zurück.

Anmerkung: Sobald das System Flag 117 auf SOFT Menü gesetzt ist, erhalten Sie über die Tastenkombination ⊡(halten) ▽, eine Liste der Funktionen aus dem aktuellen Menü. So z.B. erhalten Sie für die ersten beiden Seiten im BASE Menü folgendes:



Um die Einstellung auf CHOOSE Kästchen zurückzustellen, verwenden Sie die Tastefolge:

MODE FLAGS ▲ ← ▲ ▼ CHK OK OK.

Anmerkungen:

1. Das Menü TOOL, welches Sie durch Drücken der Taste TOOL, wird immer ein SOFT Menü erzeugen.
2. Die meisten Beispiele dieses Handbuchs zeigen beide Varianten SOFT Menüs und CHOOSE Kästchen. Für Programmieranwendungen (Kapitel 21 und 22) benutzen Sie ausschließlich SOFT Menüs.
3. Zusätzliche Informationen zu SOFT Menus vs. CHOOSE Kästchen erhalten Sie in Kapitel 2 dieses Handbuchs.

Das Menü TOOL

Die Funktionstasten für das gegenwärtig angezeigte Menü, bekannt als das TOOL Menü, sind Operationen zur Manipulation von Variablen zugeordnet (siehe zusätzliche Seiten über Variablen):

EDIT	F1	EDIT – bearbeitet den Inhalt einer Variablen (siehe Kapitel 2 und Anhang L für weitere Informationen zu diesem Thema)
VIEW	F2	VIEW – zeigt den Inhalt einer Variablen
RCL	F3	ReCaLl – holt den Inhalt der Variablen zurück
STO	F4	STOre – speichert den Inhalt einer Variablen
PURGE	F5	PURGE – löscht (bereinigt) eine Variable

CLEAR F6 CLEAR – löscht das Displays oder den Stack

Der Rechner hat nur insgesamt sechs Funktionstasten, deshalb können jeweils nur 6 Schaltflächen auf einmal angezeigt werden. Ein Menü kann aber auch mehr als nur sechs Einträge besitzen. Eine Gruppe von 6 Einträgen wird als Menüseite bezeichnet. Eigentlich hat das TOOL Menü acht Einträge, aufgeteilt auf zwei Seiten. Die zweite Seite, in welchem die letzten beiden Einträge enthalten sind, kann durch drücken der Taste NXT (Menü NeXT – nächstes Menü) angezeigt werden. Auf der Tastatur ist dies die dritte Taste von links in der dritten Reihe von unten.
In diesem Fall sind nur den ersten beiden Funktionstasten Befehle zugeordnet. Diese Befehle lauten:

CASCM F1 CASCMD: Der Befehl CAS CoMmanD, welcher dazu benutzt wird einen Befehl aus dem CAS aus einer Liste zu starten

HELP F2 Die Hilfefunktion zur Beschreibung der zur Verfügung stehenden Befehle, kann über die Funktionstaste NXT erreicht werden und wird das Original TOOL Menü anzeigen. Eine weitere Möglichkeit zum TOOL Menü zurückzukehren ist durch drücken der Taste TOOL (zweite Reihe von oben, dritte Taste von links).

Datum und Uhrzeit einstellen

Der Rechner besitzt eine interne Echtzeituhr. Diese Uhr kann im Display ständig angezeigt werden und kann sowohl für Alarme oder für geplante laufende Aufgaben eingesetzt werden. Dieser Abschnitt zeigt nicht nur wie Sie Uhrzeit und Datum einstellen können, sondern auch die Grundlagen für die Anwendung von CHOOSE Kästchen und Dateneingabe in eine Dialogbox. Die Dialogboxen Ihres Rechners, ähneln denen eines Computers.

Um Datum und Uhrzeit einzustellen verwenden wir das CHOOSE Kästchen TIME, alternativ zur Funktion der Taste 9. Kombinieren wir die rote rechte Shift-Taste ↱, mit der Taste 9, wird das CHOOSE Kästchen TIME aktiviert. Dieser Vorgang kann auch als ↱ TIME dargestellt werden. Die Abbildung unten zeigt das CHOOSE Kästchen für TIME:

Wie oben aufgeführt, stellt das Menü TIME vier verschiedene Optionen, durchnummeriert von 1 bis 4, zur Verfügung. An dieser Stelle ist für uns nur Option *3. Set time, date...* (Datum und Uhrzeit einstellen) von Interesse. Mit Hilfe der Pfeiltaste ▼ heben Sie diese Option hervor und drücken anschließend die Funktionstaten OK F6 . Anschließend erhalten Sie die Eingabemaske (siehe Anhang 1-A) zur Einstellung von Uhrzeit und Datum:

Uhrzeit einstellen

Mit den Zahlentasten, 1 2 3 4 5 6 7 8 9 0 , können Sie mit dem einstellen der Stunden beginnen. Angenommen wir ändern die Uhrzeit auf 11, indem wir 1 1 drücken, sobald das Feld SET TIME AND DATE hervorgehoben ist. Die Zahl 11 erscheint in der unteren Zeile der Eingabemaske.

Um die Änderung vorzunehmen, drücken Sie die Funktionstaste OK F6 . Im Stundenfeld erscheint nun die *11*, gleichzeitig wird das Minutenfeld hervorgehoben:

Ändern wir nun das Minutenfeld auf 25 durch drücken der Tasten : 2 5 OK . Nun wird das Sekundenfeld hervorgehoben. Angenommen Sie möchten dieses Feld auf 45 ändern, geben Sie ein: 4 5 OK

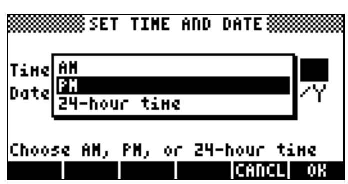
Nun wird Feld für das Zeitformat hervorgehoben. Um die aktuellen Einstellungen des Feldes zu ändern, können Sie entweder die Taste +/- (zweite Taste von links, fünfte von unten) oder die Funktionstaste CHOOSE (F2) drücken.

- Benutzen Sie die Taste +/-, wird sich das Feld für das Zeitformat in eine der nachfolgenden Optionen ändern.

    - AM : zeigt an, dass es sich um eine Uhrzeit vor Mittag handelt,d.h. AM
    - PM : zeigt an, dass es sich um eine Uhrzeit nach Mittag handelt, d.h. PM
    - 24-hr: zeigt an, dass die Uhrzeit im 24 Stundenformat angezeigt wird wobei z.B. 18:00 h, 6pm darstellt

  Die zuletzt ausgewählte Option wird, sobald Sie nachfolgende Prozedur anwenden, eingestellt.

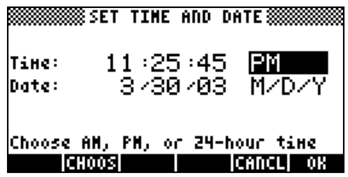
- Wenn Sie dabei die Funktionstaste CHOOS benutzen, stehen Ihnen folgende Optionen zur Auswahl:



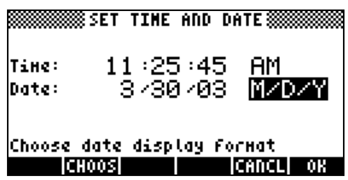
Benutzen Sie die Pfeiltasten, ▲ ▼, um zwischen diesen drei Optionen (AM, PM und 24-h) auszuwählen. Um Ihre Auswahl zu bestätigen, drücken Sie die Funktionstaste OK F6 .

Einstellen des Datums

Nachdem Sie das Format der Uhrzeit ausgewählt haben, wird die Eingabemaske SET TIME AND DATE wie folgt aussehen:



Um das Datum einzustellen, müssen Sie erst das Datumformat auswählen. Das standardmäßig eingestellte Format lautet M/D/Y (Monat/Tag/Jahr). Um dieses Format zu ändern, drücken Sie die Pfeiltaste nach unten. Das Datumsformat wird wie unten angezeigt hervorgehoben:

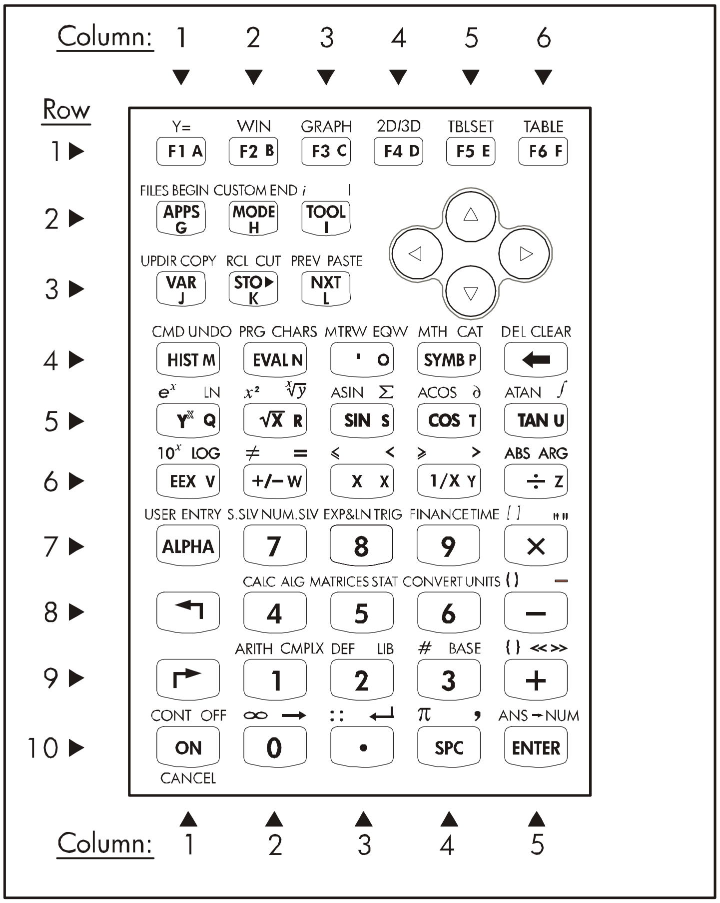


Benutzen Sie die Funktionstaste CHOOSE (B), um die verschiedenen Optionen für das Datumsformat anzeigen zu lassen:

Mit den Pfeiltasten ▲ ▼, heben Sie die gewünschte Auswahl hervor und drücken anschließend die Funktionstaste OK F6 , um diese abzuschließen.

Einführung in die Tastatur des Rechners

Die nachfolgende Abbildung zeigt ein Diagramm der Tastatur Ihres Rechners mit numerierten Zeilen und Spalten.



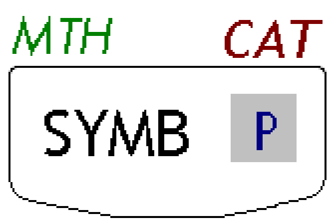
Die Abbildung zeigt 10 Reihen von Tasten, kombiniert mit 3, 5 oder 6 Spalten. Reihe 1 hat 6 Tasten, Reihe 2 und 3, jeweils 3 Tasten, während

Reihe 4 bis 10 jeweils 5 Tasten aufweisen. In der rechten oberen Ecke, in Höhe der Reihen 2 und 3, befinden sich 4 Pfeiltasten.

Jeder einzelne Taste hat drei, vier oder fünf verschiedene Funktionen. Die Hauptfunktion der Taste entspricht der auf der Tastatur hervorgehobenen (größten) Beschriftung. Auch die grüne linke *Shift-Taste, Taste (8,1),* die rote rechte *Shift-Taste, Taste (9,1)* und die blaue ALPHA-*Taste, Taste (7,1)*, können mit anderen Tasten kombiniert werden, um alternative Funktionen, die auf der Tastatur angezeigt werden, zu starten. So hat z.B. die [SYMB] Taste, *Taste (4,4),* die folgenden sechs Funktionen, die dieser wie folgt zugeordnet sind:

[SYMB]	Hauptfunktion, starten des Menüs SYMB (SYMBolic – symbolisch)
[←] MTH	Linke-Shift Funktion zum starten des MTH (Math) Menüs
[→] CAT	Rechte-Shift Funktion zum starten der Funktion CATalog (Katalog)
[ALPHA] [P]	ALPHA Funktion, um den Großbuchstaben P einzufügen
[ALPHA] [←] [P]	ALPHA Linke-Shift Funktion, um den Kleinbuchstaben P einzufügen
[ALPHA] [→] [P]	ALPHA rechte-Shift Funktion, um das Symbol P einzufügen

Von den sechs dieser Taste zugeordneten Funktionen, werden nur vier davon auf der Taste selbst angezeigt. So sieht Ihre Tastenanzeige auf der Tastatur aus:



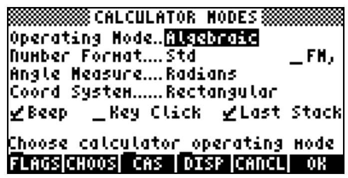
Beachten Sie Farbe und Position der Beschriftung auf der Taste, und zwar **SYMB**, *MTH*, *CAT* und **P,** zeigen die Hauptfunktion (**SYMB**) an und welche der drei weiteren Funktionen der jeweiligen Tastenkombination zugeordnet ist: linke-Shift Taste [←] (*MTH*), rechte-Shift Taste [→] (*CAT*), und [ALPHA] (**P**).

Mehr Informationen zur Tastatur des Rechners finden Sie in Anhang B.

Auswahl der Rechner Modi

In diesem Abschnitt wird vorausgesetzt, dass Sie mindestens teilweise im Umgang mit Auswahl- und Dialogboxen vertraut sind (sollten Sie es nicht sein, lesen Sie erst Kapitel 2 in diesem Handbuch nach).

Drücken Sie die Schaltfläche MODE (zweite Taste von links, zweite Reihe von oben), um die folgende CALCULATOR MODES (Rechner-Modi) Eingabemaske zu erhalten:



Drücken Sie die Funktionstaste OK F6, um zur Normalanzeige zurückzukehren. Nachfolgend einige Beispiele, wie verschiedene Rechner-Modi ausgewählt werden können.

Operationsmodus

Der Rechner bietet zwei verschieden Operationsmodi: den *algebraischen* (ALG) Modus und den *Reverse Polish Notation* (RPN) Modus. Standardmäßig ist der algebraische Modus (wie in der obigen Abbildung gezeigt) eingestellt, aber Anwender früherer HP-Rechner sind wahrscheinlich mit dem RPN-Modus vertrauter.

Um einen Operationsmodus auszuwählen, öffnen Sie zuerst die CALCULATOR MODES Eingabemaske durch drücken der Schaltfläche MODE. Das Feld *Operating Mode* (Operationsmodus) wird hervorgehoben. Wählen Sie nun *ALG* oder *RPN* Modus entweder durch drücken der Taste +/- (zweite von Links, Reihe fünf von unten) oder durch drücken der Funktionstaste CHOOSE (F2). Sollten Sie letztere zur Auswahl drücken, benutzen Sie die Pfeiltasten, ▲ ▼, zur Auswahl des entsprechenden Modus und drücken anschließend die Funktionstaste OK, um den Vorgang abzuschließen.

Um den Unterschied zwischen diesen beiden Operationsmodi zu veranschaulichen, berechnen wir nachfolgenden Ausdruck in beiden Modi:

$$\sqrt{\frac{3 \cdot \left(5 - \frac{1}{3 \cdot 3}\right)}{23^3} + e^{2.5}}$$

Um diesen Ausdruck in den Rechner einzugeben, verwenden wir zuerst den *Equation Writer (Gleichungsschreiber)*, ↱ EQW . Beachten Sie nachfolgende Tasten, neben den numerischen Tasten des Tastenfeldes:

↰ ↱ • SPC × + − ÷ √x
Yˣ eˣ ( ) ↱ EQW ◄ ► ▼ ▲ ENTER

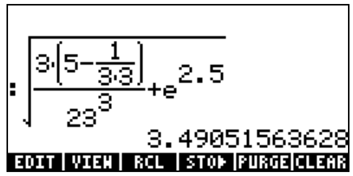
Der EquationWriter ist ein Anzeigemodus, in welchem Sie mathematische Ausdrücke über explizite mathematische Bezeichnung, einschließlich Brüche, Ableitungsfunktionen, Integrale, Wurzeln, usw., bilden können. Um mit dem EquationWriter den oben angegebenen Ausdruck eingeben zu können, benutzen Sie die Tastenfolge:

↱ EQW √x 3 × ↰ ( ) 5 −
1 ÷ 3 × 3
▲ ▲ ▲ ▲ ▲ ▲ ▲
÷ 2 3 Yˣ 3 ► ► + ↰ eˣ 2 • 5 ENTER

Nachdem Sie die Taste ENTER drücken wird im Display des Rechners, nachfolgender Ausdruck angezeigt:

```
√(3.*(5.-1/(3.*3.))/(23.^3+EXP(2.5))
```

Durch erneutes Drücken der Taste ENTER, wird folgender Wert ausgegeben; bei Anfrage (akzeptieren Sie Approx. mode (ungefährer Modus) ein, indem Sie OK drücken) [Anmerkung: Die oben verwendeten Integerwerte, z.B. 3, 5, 1 stellen exakte Werte dar. EXP(2,5) hingegen, kann nicht als exakter Wert dargestellt werden; somit ist ein Umschalten in den Approx (Näherungs)-Modus notwendig]:



Sie könnten den Ausdruck aber auch direkt, ohne den EquationWriter wie folgt eingeben:

[√x] [←] [()] [3] [.] [×] [←] [()] [5] [.] [−]
[1] [.] [÷] [←] [()] [3] [.] [×] [3] [.] [▶] [▶]
[÷] [2] [3] [.] [y^x] [3] [+] [←] [e^x] [2] [.] [5] [ENTER]

um das gleiche Ergebnis zu erzielen.

Ändern Sie nun den Modus auf RPN indem Sie zuerst die Schaltfläche [MODE] drücken. Wählen Sie den *RPN*-Modus entweder über die Taste [+/-] oder durch drücken der Funktionstaste [CHOOSE]. Drücken Sie die Funktionstaste [OK] [F6], um den Vorgang abzuschließen. Die Anzeige für den RPN-Modus sieht wie folgt aus:

Beachten Sie, dass im Display die verschiedenen Ausgabe-Ebenen, von unten nach oben mit 1, 2, 3, usw. durchnummeriert, angezeigt werden. Dies ist der sogenannte *Stack* (*Stapel*) des Rechners. Die verschiedenen Ebenen werden als *Stack Ebenen* bezeichnet , d.h. Stack Ebene 1, Stack Ebene 2, usw.

Grundsätzlich bedeutet RPN dass Sie, wenn Sie z.B. die Operation 3 + 2 in den Rechner eingeben wollen nicht die Tastenfolge [3] [+] [2] [ENTER], benutzen, sondern erst die Operanden ,in der richtigen Reihenfolge, dann erst den Operator , d.h. [3] [ENTER] [2] [ENTER] [+] eingeben.
Während Sie die Operanden , eingeben, erscheinen diese in unterschiedlichen Stack Ebenen. Geben Sie [3] [ENTER] ein, erscheint die 3 in Stack Ebene 1. Geben Sie als nächstes [2] [ENTER] ein, wird die 3 eine Ebene

nach oben, d.h. in Stack Ebene 2 verschoben. Geben Sie schließlich [+] ein, sagen wir dem Rechner, dass er den Operator oder das Programm [+] auf die Objekte aus Stack Ebene 1 und 2 anwenden soll. Das Ergebnis, 5, wird in Stack Ebene 1 angezeigt. Ein einfacherer Weg diese Operation zu berechnen ist: [3][ENTER][2][+].

Versuchen wir ein paar weitere einfache Operationen, bevor wir uns an die komplizierteren, aus dem vorangegangenen algebraischen Modus wenden:

123/32	[1][2][3][ENTER][3][2][÷]
4^2	[4][ENTER][2][y^x]
$^3\sqrt{27}$	[2][7][ENTER][3][↱] $\sqrt[x]{y}$

Beachten Sie die Position von y und x in den letzten beiden Operationen. Bevor die Taste [y^x] gedrückt wird, ist die Basis der Exponential-Operation y (Stack Ebene 2) während der Exponent x (Stack Ebene 1) ist. Ähnlich verhält es sich mit der Quadratwurzel, y (Stack Ebene 2) ist die Zahl unter dem Wurzelzeichen während x (Stack Ebene 1) die Wurzel selbst darstellt.

Versuchen Sie folgendes Beispiel, indem Sie 3 Faktoren einbeziehen:
$(5 + 3) \times 2$

[5][ENTER][3][+]	Berechnet zuerst den Wert (5+3).
[2][×]	Vervollständigt die Berechnung.

Versuchen wir nun den weiter oben vorgeschlagenen Ausdruck:

$$\sqrt{\frac{3 \cdot \left(5 - \frac{1}{3 \cdot 3}\right)}{23^3} + e^{2.5}}$$

[3][•][ENTER]	Geben Sie 3 in Ebene 1 ein
[5][•][ENTER]	Geben Sie 5 in Ebene 1 ein, 3 wird nach y verschoben
[3][•][ENTER]	Geben Sie 3 in Ebene 1 ein, 5 wird in Ebene 2 und 3 in Ebene 3 verschoben

[3][•][×]	Setzen Sie 3 und das Multiplikationszeichen, 9 erscheint in Ebene 1
[1/x]	1/(3x3), letzter Wert in Ebene 1; 5 in Ebene 2, 3 in Ebene 3
[−]	5 - 1/(3×3) belegt nun Ebene 1, 3 ist in Ebene 2
[×]	3× (5 - 1/(3×3)) belegt nun Ebene 1.
[2][3][•][ENTER]	Geben Sie 23 in Ebene 1 ein, 14,66666 wird in Ebene 2 verschoben.
[3][•][Y^x]	Geben Sie 3 ein, berechnen Sie 23^3 in Ebene 1. 14,666 befindet sich in Ebene 2.
[÷]	$(3\times (5\text{-}1/(3\times 3)))/23^3$ in Ebene 1
[2][•][5]	Geben Sie 2,5 in Ebene 1 ein
[↰] e^x	$e^{2,5}$, erscheint in Ebene 1, in Ebene 2 wird der vorangegangene Wert angezeigt.
[+]	$(3\times (5 - 1/(3\times 3)))/23^3 {}_{+} e^{2,5}$ = 12.18369, in Ebene 1.
[√x]	$\sqrt{}((3\times (5 - 1/(3\times 3)))/23^3 {}_{+} e^{2,5})$ = 3,4905156, in Ebene 1.

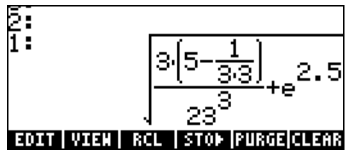
Obwohl im RPN-Modus mehr denken als im ALG-Modus erforderlich ist, gibt es vielfache Vorteile in der Verwendung des RPN-Modus. So z.B. können Sie im RPN-Modus sehen, wie sich die Gleichung Schritt für Schritt entfaltet. Dies ist äußerst nützlich um mögliche Eingabefehler aufzufinden. Sobald Sie auch effizienter in diesem Modus werden und einige Tricks herausfinden, werden Sie den gleichen Vorgang schneller und mit weniger Tastenanschlägen berechnen können. Nehmen wir z.B. die Berechnung von (4×6 - 5)/(1+4×6 - 5). Im RPN-Modus können Sie wie folgt eingeben:

[4] [ENTER] [6] [×] [5] [−] [ENTER] [1] [+] [÷]

offensichtlich, können Sie auch im RPN-Modus, einen Ausdruck in der gleichen Reihenfolge wie im algebraischen Modus eingeben, indem Sie den EquationWriter benutzen. So zum Beispiel:

[↱] EQW [√x][3][•][×][↰] ( ) [5][•][−][1][÷][3][•][×][3][•]
[▲][▲][▲][▲][▲][▲][▲]
[÷][2][3][•][Y^x][3][▶][▶][+][↰] e^x [2][•][5][ENTER]

Der daraus resultierende Ausdruck wird in Stack Ebene 1, wie folgt angezeigt:



Beachten Sie, dass der Ausdruck in Stack Ebene 1 nach drücken der Taste ENTER erscheint. Drücken Sie jedoch die Taste EVAL an dieser Stelle, wird der numerische Wert des Ausdrucks berechnet. Beachten Sie: Wenn Sie im RPN-Modus die Taste Enter drücken, sofern keine Kommandozeile vorhanden ist, wird dieser Befehl die DUP Funktion auslösen, welche den Inhalt der Stack Ebene 1 eine Ebene nach oben, also nach 2 kopiert (genauso werden alle weiteren Ebenen eine Zeile nach oben verschoben). Dies ist sehr nützlich, wie Sie in den vorangegangenen Beispielen sehen konnten.

Um den Modus ALG vs. RPN auszuwählen, können Sie auch System-Flag 95 mit folgender Tastenfolge löschen/bereinigen:

MODE FLAGS ▲ ↰ ▲ ↰ ▲ ↰ ▲ CHK OK OK

Alternativ dazu können Sie eine der folgenden Abkürzungen verwenden:

- Im ALG-Modus,
  CF(-95) wählt den RPN-Modus aus

- Im RPN-Modus,
  95 +/- ENTER SF wählt den ALG-Modus aus

Weitere Informationen zu den System Flags des Rechners, finden Sie in Kapitel 2.

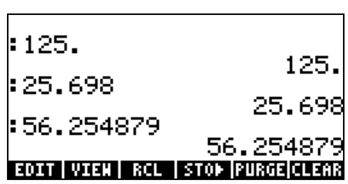
Zahlenformat und Dezimalpunkt oder Komma

Das Wechseln des Zahlenformates erlaubt Ihnen die benutzerspezifische Anpassung, reelle Zahlen im Rechner anzuzeigen. Sie werden sehen, dass dies in Operationen mit Zehnerpotenzen äußerst nützlich ist, oder aber um die Dezimalstellen in einem Ergebnis einzuschränken.

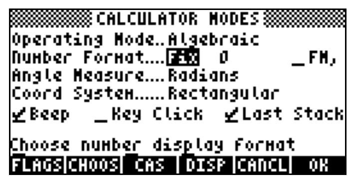
Um ein Zahlenformat auszuwählen, öffnen Sie zuerst die CALCULATOR MODES Eingabemaske durch drücken der Schaltfläche MODE. Benutzen Sie anschließend die Pfeiltaste ▼, um die Option *Number format* (Zahlenformat) auszuwählen. Der voreingestellte Wert ist *Std* oder *Standard* format. Im Standardformat werden Gleitpunktzahlen mit maximaler Genauigkeit (12 Wertziffern - Nachkommastellen), die der Rechner erlaubt, angezeigt. Mehr über reelle Zahlen finden Sie in Kapitel 2. Um dieses und weitere Zahlenformate zu veranschaulichen, machen Sie die folgenden Übungen:

- Standardformat:
  Dies ist der am häufigsten verwendete Modus, weil dieser Zahlen in der gängigsten Schreibweise anzeigt.
  Drücken Sie die Funktionstaste OK mit dem *Number format* (Zahlenformat) auf *Std* eingestellt, um dann zum Display des Rechners zurückzukehren. Geben Sie die Zahl 123,4567890123456 ein. Beachten Sie, dass diese Zahl 16 Nachkommastellen beinhaltet. Drücken Sie die Taste ENTER. Die Zahl wird auf maximal 12 Wertziffern abgerundet und wird wie folgt angezeigt:

In der Dezimalanzeige werden Integer Zahlen trotz allem ohne Dezimalnullen angezeigt. Zahlen mit unterschiedlichen Nachkommastellen, werden im Display so ausgerichtet, dass nur die Dezimalstellen, die nötig sind, angezeigt werden. Nachfolgend weitere Beispiele von Zahlen im Standardformat:



- Feststehendes Format ohne Dezimalzahlen: Drücken Sie die Schaltfläche MODE. Benutzen Sie anschließend die Pfeiltaste ˅, um die Option *Number format* (Zahlenformat) auszuwählen. Drücken Sie die Funktionstaste CHOOSE (F2) und wählen Sie die Option *Fix* mit der Pfeiltaste ˅.
-

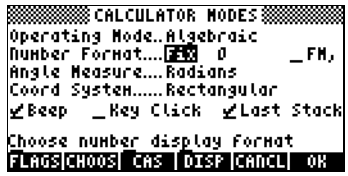


Beachten Sie, dass das Zahlenformat auf *Fix*, gefolgt von einer Null *(0)* gesetzt ist. Diese Zahl zeigt die Anzahl der Dezimalstellen, welche nach dem Dezimalkomma im Display des Rechners angezeigt werden sollen. Drücken Sie die Funktionstaste OK, um zum Display des Rechners zurückzukehren. Die Zahl wird nun wie folgt angezeigt:

Durch diese Einstellung werden alle Ergebnisse zu der nächsten Integer Zahl abgerundet erzwungen (keine Nachkommastelle wird angezeigt). Im Rechner selbst, wird diese Zahl mit allen 12 Nachkommastellen gespeichert. Je nachdem, wie wir die Anzahl der anzuzeigenden Dezimalstellen ändern, werden zusätzliche Zahlen erneut sichtbar.

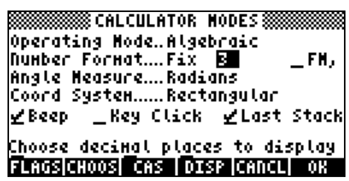
- Feststehendes Format mit Dezimalzahlen:
  Dieser Modus wird bei der Arbeit mit begrenzter Präzision benutzt. So z.B., wenn Sie finanzmathematische Berechnungen, im FIX 2 Modus durchführen, ist es bequem, da man ganz einfach Währungseinheiten bis zu einer Präzision von 1/100 darstellen kann.
  Drücken Sie die Schaltfläche MODE. Benutzen Sie anschließend die Pfeiltaste ˅, um die Option *Number format* (Zahlenformat)

auszuwählen. Drücken Sie die Funktionstaste CHOOSE (F2) und wählen Sie die Option *Fix* mit der Pfeiltaste ▼.



Drücken Sie die Pfeiltaste, ▶, um die Null vor der Option *Fix* hervorzuheben. Drücken Sie anschließend die Funktionstaste CHOOSE und wählen Sie mit den Pfeiltasten ▲▼, sagen wir 3 Dezimalstellen aus.

Drücken Sie die Funktionstaste OK, um die Auswahl abzuschließen.

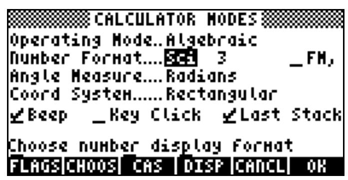


Drücken Sie die Funktionstaste OK, um zum Display des Rechners zurückzukehren. Die Zahl wird nun wie folgt angezeigt:

Beachten Sie, dass die Zahl nun abgerundet und nicht abgeschnitten ist. Somit wird die Zahl 123,4567890123456 in dieser Einstellung als

123,457 und nicht als 123,456 angezeigt, da die Nachkommastelle nach der 6 > 5 ist.

- Wissenschaftliches Format
  Das wissenschaftliche Format wird hauptsächlich zum Lösen von Problemen in der Physik, wo Zahlen gewöhnlich mit begrenzter Präzision, multipliziert mit einer Zehnerpotenz, angezeigt werden, benutzt.
  Zum Einstellen dieses Formates, drücken Sie die Schaltfläche MODE. Benutzen Sie anschließend die Pfeiltaste ▼, um die Option *Number format* (Zahlenformat) auszuwählen. Drücken Sie die Funktionstaste CHOOSE (F2) und wählen Sie die Option *Scientific* (wissenschaftlich) mit der Pfeiltaste ▼. Behalten Sie die Zahl 3 vor *Sci* . (Diese Zahl kann genauso geändert werden, wie wir dies im *Fixed* Zahlenformat mit den Dezimalstellen aus obigem Beispiel abgeändert haben.)

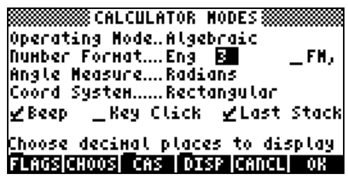


Drücken Sie die Funktionstaste OK, um zum Display des Rechners zurückzukehren. Die Zahl wird nun wie folgt angezeigt:

Dieses Ergebnis, 1,23E2, ist die Darstellung einer Zehnerpotenz des Rechners, d.h. $1,235 \times 10^2$. In dieser sogenannten wissenschaftlichen Schreibweise, stellt die dem *Sci* Zahlenformat vorangestellte Ziffer 3 (wie vorher gezeigt) die Anzahl der Dezimalstellen nach dem Komma dar. Die wissenschaftliche Darstellung beinhaltet immer eine Ganzzahl, wie oben zu sehen ist. Deshalb ist in diesem Fall die Anzahl der Nachkommastellen vier.

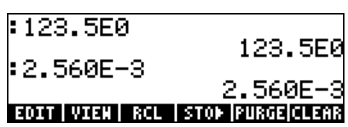
Technisches Format

Das technische Format ähnelt sehr dem wissenschaftlichen Format, mit der Ausnahme, dass die Zehnerpotenzen ein Vielfaches von drei sind.
Zum Einstellen dieses Formates, drücken Sie die Schaltfläche MODE. Benutzen Sie anschließend die Pfeiltaste ▼, um die Option *Number format* (Zahlenformat) auszuwählen. Drücken Sie die Funktionstaste CHOOSE (F2) und wählen Sie mit der Pfeiltaste ▼ die Option *Engineering* (technisch). Behalten Sie die Zahl 3 vor dem *Eng*. (Diese Zahl kann genauso geändert werden, wie wir dies im *Fixed* Zahlenformat mit den Dezimalstellen aus vorangegangenem Beispiel abgeändert haben.)



Drücken Sie die Funktionstaste OK, um zum Display des Rechners zurückzukehren. Die Zahl wird nun wie folgt angezeigt:

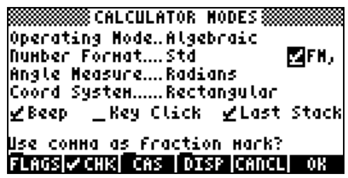
Da diese Zahl drei Ziffern im Integer-Teil enthält, wird diese im technischen Format mit vier Wertstellen und einer Zehnerpotenz von Null angegeben. So z.B. wird die Zahl 0,00256 wie unten angezeigt:



Dezimalkomma vs. Dezimalpunkt

Dezimalpunkte in Gleitpunktzahlen können durch ein Komma ersetzt werden, wenn der Benutzer mit diesen besser vertraut ist. Um Dezimalpunkte durch Kommas zu ersetzen, ändern Sie die Option *FM* in der CALCULATOR MODES Eingabemaske, wie folgt auf Kommas (Beachten Sie, dass wir das *Zahlenformat* auf *Std* geändert haben):

- Drücken Sie die Schaltfläche MODE. Drücken Si anschließend die Pfeiltaste ▼, einmal und die Pfeiltaste nach rechts, ▶, zweimal, um die Option __*FM* hervorzuheben. Um Kommas auszuwählen, drücken Sie die Funktionstaste CHECK (d.h. die Taste F2). Die Eingabemaske sieht wie folgt aus:
-



-
- Drücken Sie die Funktionstaste OK, um zum Display des Rechners zurückzukehren. Die Zahl 123,456789012, die Sie vorher eingegeben haben, wird nun wie folgt angezeigt:
-

Winkelmaß

Trigonometrische Funktionen benötigen z.B. Argumente, die Flächenwinkel darstellen. Der Rechner stellt drei verschiedene *Winkelmaß*-Modi für die Arbeit mit Winkeln zur Verfügung, und zwar:

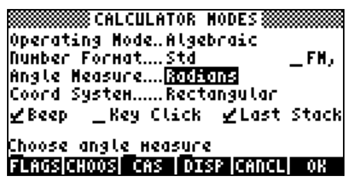
- *Grade*: Ein kompletter Umfang beträgt *360* Grad (*360°*) oder *90* Grad (*90°*) in einem rechten Winkel. Diese Darstellung wird hauptsächlich in der Standardgeometrie, Maschinen- oder Stahlbau und im Vermessungswesen eingesetzt.
- *Radiane*: In einem kompletten Umfang sind 2π Radiane ($2\pi\ ^r$) enthalten oder $\pi/2$ Radiane ($\pi/2\ ^r$) in einem rechten Winkel. Diese Notation wird hauptsächlich bei der Lösung mathematischer oder physischer Probleme verwendet. Dies ist der Standard-Modus des Rechners.
- *Zentesimalgrade*: Ein kompletter Umfang beträgt *400* Zentesimalgrade ($400\ ^g$) oder *100* Zentesimalgrade ($100\ ^g$) in einem rechten Winkel. Diese Notation ist ähnlich dem Grad-Modus und war eigentlich zur

"Vereinfachung" der Grad-Notation gedacht, wird jedoch heute selten benutzt.

Das Winkelmaß beeinflusst die trigonometrischen Funktionen wie SIN, COS, TAN und damit verbundene Funktionen.

Um den Winkelmaß-Modus zu wechseln, gehen Sie wie folgt vor:

- Drücken Sie die Schaltfläche MODE. Drücken Sie anschließend die Pfeiltaste ▼ zweimal. Wählen Sie nun den *Winkelmaß*-Modus entweder durch drücken der Taste +/- (zweite von Links, Reihe fünf von unten) oder durch drücken der Funktionstaste CHOOSE (F2). Sollten Sie sich für letztere entscheiden, benutzen Sie die Pfeiltasten ▲▼ zur Auswahl des entsprechenden Modus und drücken anschließend die Funktionstaste OK F6 , um den Vorgang abzuschließen. Im nachfolgenden Beispiel wurde der Radian-Modus gewählt:
-



Koordinatensystem

Das Koordinatensystem beeinflusst die Eingabe- und Darstellungsart von Vektoren und komplexen Zahlen. Mehr über komplexe Zahlen und Vektoren erfahren Sie in Kapitel 4 bzw. 9.

Zwei- und dreidimensionale Vektorkomponenten und komplexe Zahlen können in jedem der 3-Koordinatensysteme dargestellt werden: das Kartesische (2 dimensional) oder das rechtwinklige (3 dimensional), das zylindrische (3 dimensional) oder das Polarsystem (2 dimensional) und das sphärische (nur 3 dimensional). In einem Kartesischen oder rechtwinkliken Koordinatensystem hat ein Punkt P drei lineare Koordinaten (x,y,z), gemessen vom Ursprung entlang von 3 zueinander senkrechten Achsen (im 2-D Modus wird z als 0 angenommen). In einem zylindrischen oder einem Polarsystem werden die Koordinaten eines Punktes von (r,θ,z) bestimmt, wobei r eine radiale Distanz, gemessen vom Ursprung auf die xy Ebene darstellt, θ den Winkel, den diese radiale Distanz r mit der positiven Achse x bildet – gemessen als positive

Richtung gegen den Uhrzeigersinn – und z das gleiche wie die z-Koordinate in einem Kartesischen System darstellt (im 2-D Modus wird z als 0 angenommen). Das rechtwinklige und das Polar System sind durch die nachfolgenden Mengen verwandt (verbunden):

$$x = r \cdot \cos(\theta) \qquad r = \sqrt{x^2 + y^2}$$

$$y = r \cdot \sin(\theta) \qquad \theta = \tan^{-1}\left(\frac{y}{x}\right)$$

$$z = z$$

In einem sphärischen System werden die Koordinaten eines Punktes durch (ρ,θ,ϕ) bestimmt, wobei ρ eine radiale Distanz, gemessen vom Ursprung eines Kartesischen Systems, θ der Winkel, der den Winkel die durch die Projektion der linearen Distanz ρ auf die xy Achse darstellt (genau wie θ in Polar-Koordinaten) und ϕ den Winkel von der positiven Achse z auf die radiale Distanz ρ darstellt. Das rechtwinklige und das sphärische System sind durch die nachfolgenden Mengen miteinander verbunden:

$$x = \rho \cdot \sin(\varphi) \cdot \cos(\theta) \qquad \rho = \sqrt{x^2 + y^2 + z^2}$$

$$y = \rho \cdot \sin(\varphi) \cdot \cos(\theta) \qquad \theta = \tan^{-1}\left(\frac{y}{x}\right)$$

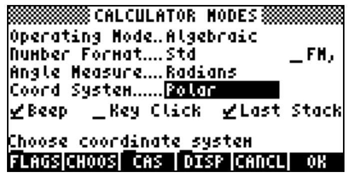
$$z = \rho \cdot \cos(\varphi) \qquad \varphi = \tan^{-1}\left(\frac{\sqrt{x^2 + y^2}}{z}\right)$$

Um das Koordinaten-System in Ihrem Rechner zu ändern, führen Sie folgende Schritte durch:

- Drücken Sie die Schaltfläche MODE. Drücken Sie anschließend die Pfeiltaste ▼ dreimal. Wählen Sie nun den *Winkelmaß*-Modus entweder durch drücken der Taste +/- (zweite von Links, Reihe fünf von unten) oder durch drücken der Funktionstaste CHOOS (F2). Sollten Sie letztere zur Auswahl drücken, benutzen Sie die Pfeiltasten ▲▼ zur Auswahl des entsprechenden Modus und drücken anschließend die Funktionstaste

OK F6, um den Vorgang abzuschließen. Im nachfolgenden Beispiel wird der Polar Koordinaten Modus gewählt:

•



Beep (Piepsen), Key Click (Tastenklick) und Last Stack (letzter Stack)

Die letzte Zeile in der CALCULATOR MODES Eingabemaske, enthält folgende Optionen:

_Beep _Key Click _Last Stack

Wählen Sie das Häkchen neben einer der Optionen aus, wird die entsprechende Option aktiviert. Diese Optionen werden nachfolgend beschrieben.

_Beep : Wenn ausgewählt, ist der Beeper des Rechners aktiviert. Diese Operation dient hauptsächlich für Fehlermeldungen, hat aber auch einige weitere Funktionen, wie:BEEP.

_Key Click : Wird diese Funktion ausgewählt, wird bei jedem Tastenanschlag ein "Klick" hörbar.

_Last Stack: Behält den Inhalt der letzten Stack Eingabe für die Weiterverwendung mit den Funktionen UNDO und ANS (siehe Kapitel 2).

Die *_Beep* Option kann bei Fehlermeldungen für den Anwender nützlich sein. Diese Option sollten Sie ausschalten bevor Sie ein Klassenzimmer oder eine Bibliothek, in welchem Sie den Rechner benutzen möchten, betreten.
Die *_Key Click* Option kann auch als hörbare Überprüfung für die Richtigkeit der Tasteneanschläge dienen.
Die *_Last Stack* Option ist besonders dann vor Vorteil, wenn wir den letzten Vorgang rückgängig machen möchten, wenn dieser für eine weitere Berechnung benötigt wird.

Um irgendeine dieser drei Optionen zu selektieren oder zu deselektieren, drücken Sie zuerst die Taste MODE. Als nächstes

- Benutzen Sie die Pfeiltaste, ˜, viermal um die die Option *_Last Stack* auszuwählen. Verwenden Sie die Funktionstaste CHECK (d.h. die Taste F2) um die Auswahl zu ändern.
- Drücken Sie die Pfeiltaste ◄ um die Option *_Key Click* auszuwählen. Verwenden Sie die Funktionstaste CHECK (d.h. die Taste F2) um die Auswahl zu ändern.
- Drücken Sie die Pfeiltaste ◄ um die Option *_Beep* auszuwählen. Verwenden Sie die Funktionstaste CHECK (d.h. die Taste F2) um die Auswahl zu ändern.

  Drücken Sie die Funktionstaste OK F6, um den Vorgang abzuschließen.

Auswahl der CAS-Einstellungen

CAS steht für *C*omputer *A*lgebraic *S*ystem (Algebraisches System des Rechners). Dies ist das mathematische Herzstück des Rechners, in welchem die symbolischen mathematischen Operationen und Funktionen programmiert sind und auch ausgeführt werden. Das CAS bietet eine Reihe von Einstellungen, die nach Typ oder Operationsart eingestellt werden können. Dies sind:

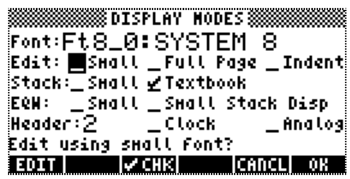
- die unabhängige Standardvariable
- numerischer vs. symbolischer Modus
- Näherungs- vs. exakter Modus
- Verbose vs. Non-verbose Modus
- Einzelschrittmodus für Operationen
- Aufsteigendes Potenzformat für Polynome
- Genauer Modus
- Vereinfachung von irrationalen Ausdrücken

Weitere Details zur Auswahl der CAS Einstellungen finden Sie in Anhang C.

Auswahl der verschiedenen Anzeige-Modi

Durch Auswahl der verschiedenen Anzeige-Modi, kann das Display des Rechners Ihren Wünschen angepasst werden. Um die möglichen Display-Einstellungen anzusehen, gehen Sie wie folgt vor:

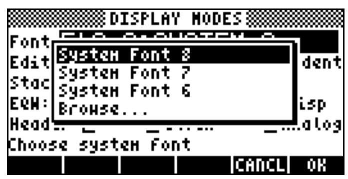
- Drücken Sie die Schaltfläche MODE um die CALCULATOR MODES Eingabemaske zu starten. Innerhalb der CALCULATOR MODES Eingabemaske, drücken Sie die Funktionstaste DISP (F4), um die Eingabemaske DISPLAY MODES anzuzeigen.
-



- Benutzen Sie die Pfeiltasten, um zwischen den einzelnen Optionen der DISPLAY MODES Eingabemaske zu navigieren: ◀▶▼▲.
- Um irgendeine der obigen Einstellungen aus- oder abzuwählen, die die Auswahl eines Häkchens benötigen, wählen Sie zuerst den Unterstrich vor der von Ihnen gewünschten Option und bewegen Sie die CHK Funktionstaste solange, bis die gewünschte Einstellung erreicht ist. Sobald eine Option gewählt wurde, erscheint ein Häkchen über dem Unterstrich (d.h. im obigen Beispiel die Option *Textbook* im *Stack*: Linie oben). Nicht ausgewählte Optionen haben kein Häkchen über dem Unterstrich vor der gewünschten Option (z.B. die Optionen *_Small, _Full page* und *_Indent* bei *Edit*: Linie oben).
- Um die Schrift für das Display auszuwählen, heben Sie das Feld vor der Option *Font*: in der DISPLAY MODES Eingabemaske vor und benutzen Sie die Funktionstaste CHOOSE (F2).
- Nachdem Sie nun alle gewünschten Optionen, welche Sie für die Eingabemaske des DISPLAY MODES haben möchten, ausgewählt oder abgewählt haben, drücken Sie die Funktionstaste OK. Somit kehren Sie zur CALCULATOR MODES Eingabemaske zurück. Um zur Normalansicht des Rechners zurückzukehren, drücken Sie die Taste OK ein weiteres Mal.

Auswahl der Schrift im Display

Durch Veränderung der Schriftgröße, können Sie den Rechner nach Ihrem ermessen einstellen. So z.B., wenn Sie eine 6-Pixel Schrift verwenden, können Sie in Ihrem Display bis zu 9 Stack Ebenen anzeigen. Folgen Sie diesen Anweisungen, um die Schrift für Ihr Display auszuwählen:
Drücken Sie die Schaltfläche MODE um die CALCULATOR MODES Eingabemaske zu starten. Innerhalb der CALCULATOR MODES Eingabemaske, drücken Sie die Funktionstaste DISP (F4), um die Eingabemaske DISPLAY MODES anzuzeigen. Das Feld *Font*: ist hervorgehoben und die Option *Ft8_0:system 8* ist ausgewählt. Dies ist die standardmäßig eingestellte Schrift. Wenn Sie nun die Funktionstaste CHOOSE (F2) drücken, erhalten Sie eine Auflistung aller im System vorhandenen Schriften, wie unten angezeigt:



Die vorhandenen Optionen sind drei Standard *System Fonts* (Größe 8, 7 und 6) und eine *Browse*.. (Such) Option. Über letztere Option können Sie den Speicher des Rechners nach weiteren Schriften durchsuchen, welche Sie eventuell selbst erzeugt (siehe Kapitel 23) oder auf Ihren Rechner draufgespielt haben.
Üben Sie, indem Sie die Schrift des Rechners von 7 auf 6 umstellen. Drücken Sie die Funktionstaste OK, um die Auswahl abzuschließen. Nachdem Sie nun eine Schrift ausgewählt haben, drücken Sie die Funktionstaste OK, um zur CALCULATOR MODES Eingabemaske zurückzukehren. Um an dieser Stelle zur Normalansicht des Rechner zurückzukehren, drücken Sie die Funktionstaste OK erneut und beachten Sie, wie sich die Stack Anzeige verändert, um sich der neuen Schriftart anzupassen.

Auswahl der Eigenschaften des Zeileneditors

Drücken Sie die Schaltfläche MODE um die CALCULATOR MODES Eingabemaske zu starten. Drücken Sie die Funktionstaste DISP (F4), innerhalb der CALCULATOR MODES Eingabemaske, um die Eingabemaske DISPLAY MODES anzuzeigen. Drücken Sie die Pfeiltaste ▼ einmal um zur *Edit* Zeile (Bearbeitungszeile) zu gelangen. Diese Zeile weist drei Merkmale auf, die verändert werden können. Sind diese Eigenschaften ausgewählt (mit einem Häkchen davor) sind folgende Effekte aktiviert:

_Small	Die Schrift wird auf kleiner geändert
_Full page	Erlaubt es den Cursor ans Zeilenende zu bewegen
_Indent	Automatischer Zeileneinzug, wenn eine Zeilenschaltung vorgenommen wird

Genaue Anweisungen zur Anwendung des Zeileneditors finden Sie in Kapitel 2 dieses Handbuchs.

Auswahl der Stack-Eigenschaften

Drücken Sie die Schaltfläche MODE um die CALCULATOR MODES Eingabemaske zu starten. Innerhalb der CALCULATOR MODES Eingabemaske, drücken Sie die Funktionstaste DISP (F4), um die Eingabemaske DISPLAY MODES anzuzeigen. Drücken Sie die Pfeiltaste ▼ zweimal um zur *Stack* Zeile zu gelangen. Diese Zeile weist zwei Einstellungen auf, die verändert werden können. Sind diese Eigenschaften ausgewählt (mit einem Häkchen davor) sind folgende Effekte aktiviert:

_Small	Die Schrift wird auf kleiner geändert. Maximiert die Anzahl der Informationen, die auf dem Display angezeigt werden. Beachten Sie, diese Auswahl überschreibt die für die Stack Anzeige ausgewählte Schriftart.
_Textbook	Zeigt mathematische Ausdrücke in grafisch mathematischer Notation (Schreibweise) an

Um diese Einstellungen zu veranschaulichen, wählen Sie entweder den algebraischen oder den PRN-Modus, benutzen Sie den EquationWriter, um folgende bestimmte Integrale einzugeben:

↱ EQW ↱ ∫ 0 ▶ ↰ ∞ ▶ ↰ e^x +/- X ▶ X ENTER

Im algebraischen Modus, wenn weder *_Small* noch *_Textbook* ausgewählt wurden, sieht die nachfolgende Ansicht für das Ergebnis dieser Eingabe wie folgt aus:

Wenn nur die Option *_Small* ausgewählt wurde, sieht die Eingabe wie folgt aus:

Ist aber die Option *_Textbook* ausgewählt (Standardwert), egal ob die Option *_Small* ausgewählt wurde oder nicht, sieht das Ergebnis der Anzeige wie folgt aus:

Auswahl der Eigenschaften für den EquationWriter (EQW) (Gleichungseditor)

Drücken Sie die Schaltfläche MODE um die CALCULATOR MODES Eingabemaske zu starten. Innerhalb der CALCULATOR MODES Eingabemaske, drücken Sie die Funktionstaste DISP (F4), um die Eingabemaske DISPLAY MODES anzuzeigen. Drücken Sie die Pfeiltaste ▼ dreimal, um zur Zeile EQW (EquationWriter = Gleichungseditor) zu gelangen. Diese Zeile weist zwei Einstellungen auf, die verändert werden können. Sind diese Eigenschaften ausgewählt (mit einem Häkchen davor) sind folgende Auswahlmöglichkeiten aktiviert:

_Small	Ändert die Schrift auf klein während Sie den EquationEditor (Gleichungseditor) benutzen
_Small Stack Disp	Zeigt eine kleine Schriftart im Stack für den den Display im Textbook (Textbuch) Stil

Genaue Anweisungen zur Benutzung des Gleichungseditors (EQW) werden an anderer Stelle in diesem Handbuch beschrieben.

So wird z.B. die oben dargestellte Integrale, $\int_0^{\infty} e^{-X}\,dX$, wenn Sie *_Small Stack Disp im EQW in der Eingabemaske des DISPLAY MODES* auswählen, wie folgt dargestellt:

Auswahl der Größe für die Kopfzeile

Drücken Sie die Schaltfläche MODE um die CALCULATOR MODES Eingabemaske zu starten. Innerhalb der CALCULATOR MODES Eingabemaske, drücken Sie die Funktionstaste DISP (F4), um die Eingabemaske DISPLAY MODES anzuzeigen. Drücken Sie die Pfeiltaste ˜viermal, um zur Zeile *Header* (Kopfzeile) zu gelangen. Dem Feld *Header* ist standardmäßig der Wert 2 zugewiesen. Das bedeutet, dass der obere Teil des Displays zwei Zeilen enthält, eine in der die aktuellen Einstellungen des Rechners angezeigt werden und eine zweite in der das aktuelle Unterverzeichnis, im Speicher des Rechners angezeigt wird (Diese Zeilen wurden in einem vorangegangenen Abschnitt des Handbuchs beschrieben). Der Anwender kann diese Einstellungen auf 1 oder 0 setzen, um die Anzahl der Kopfzeilen im Display zu verringern.

Auswahl der Anzeige für die Uhr

Drücken Sie die Schaltfläche MODE um die CALCULATOR MODES Eingabemaske zu starten. Innerhalb der CALCULATOR MODES Eingabemaske, drücken Sie die Funktionstaste DISP (F4), um die Eingabemaske DISPLAY MODES anzuzeigen. Drücken Sie die Pfeiltaste ▼ viermal, um zur Zeile *Header* (Kopfzeile) zu gelangen. Das Feld *Header* (Kopfzeile) wird hervorgehoben. Benutzen Sie die rechte Pfeiltaste (▶), um den Unterstrich vor der Option *_Clock* or *_Analog* auszuwählen. Drücke Sie die Funktionstaste CHK solange, bis die gewünschte Einstellung erreicht ist. Ist die Auswahl *_Clock* hervorgehoben, wird die Uhrzeit und das Datum in der rechten oberen Ecke des Displays angezeigt. Ist auch die Option *_Analog* ausgewählt, wird eine analoge, anstelle einer digitalen Uhr angezeigt. Ist die Option _*Clock* nicht ausgewählt, oder die Kopfzeile nicht sichtbar oder zu klein, wird das Datum und die Uhrzeit im Display nicht angezeigt.

Kapitel 2
Einführung in den Rechner

In diesem Kapitel werden eine Anzahl von Basisoperationen des Rechners erläutert, einschließlich der Anwendung des EquationWriters und der Manipulation von Datenobjekten im Rechner. Studieren Sie die Beispiele in diesem Kapitel genau, um die Möglichkeiten des Rechners für zukünftige Anwendungen genau zu begreifen.

Rechner-Objekte

Jede Zahl, Ausdruck, Zeichen, Variable, usw. die im Rechner erstellt oder manipuliert werden kann, wird als Objekt bezeichnet. Einige der nützlichsten Objekttypen werden nachfolgend aufgelistet.

Reelle Zahlen. Diese Objekte stellen eine positive oder negative Zahl, mit 12 Nachfommastellen und einem Exponenten zwischen -499 und +499 dar. Beispiele von reellen Zahlen sind: 1, -5, 56,41564 1,5E45, -555,74E-95

Bei der Eingabe einer reellen Zahl können Sie die Taste EEX zum eintragen des Exponenten und die Taste +/- zum ändern des Vorzeichens des Exponenten oder der Mantisse benutzen.

Beachten Sie, dass die reelle Zahl immer mit einem Dezimalkomma eingegeben werden muss, auch wenn die Zahl keine Nachkommastellen besitzt. Andernfalls wird die Zahl als Integer betrachte, was aber ein anderes Objekt des Rechners darstellt. Reelle Zahlen verhalten sich, wie Sie es von einer Zahl in einer mathematischen Operation erwarten würden.

Integer Zahlen. Diese Objekte stellen Integer Zahlen dar (Ganzzahlen, ohne Dezimalstellen) und haben keine Grenzen (ausgenommen die des Speichers im Rechner). Beispiele von Integer Zahlen sind: 1, 564654112, -4131654673546467654654 87. Beachten Sie, dass diese Zahlen kein Dezimalkomma enthalten.

Wegen ihres Speicherformats, behalten Integerzahlen immer deren volle Genauigkeit bei der Berechnung. So z.B. wird die Operation 30/14 mit Integer Zahlen durchgeführt, erhalten Sie das Ergebnis 15/7 und nicht

2,142.... Um ein Ergebnis als Realzahl (oder Gleitkomma) zu erzwingen, benutzen Sie die Funktion →NUM ↱ →NUM .

Wird im CAS (siehe Anhang C) der APPROX (Näherungs) ausgewählt, werden Integer automatisch in reelle Zahlen umgewandelt. Sollten Sie nicht planen das CAS zu benutzen, wäre es ratsam gleich in den Näherungsmodus zu wechseln. Mehr dazu in Anhang C.

Häufig werden Integer mit reellen Zahlen gemischt oder eine Integer als reelle Zahl interpretiert. Der Rechner wird eine derartige Vermischung von Objekten feststellen und Sie fragen, ob Sie in den Näherungsmodus wechseln wollen.

Komplexe Zahlen , sind erweiterte reelle Zahlen , welche die Einheit imaginäre Zahl , $i^2 = -1$ enthalten. So wird z.B. $3 + 2i$ eine komplexe Zahl , in den Rechner als (3, 2) eingegeben.

Komplexe Zahlen können Kartesisch oder im Polarmodus, abhängig von den Einstellungen des Rechners, dargestellt werden. Beachten Sie dabei, dass komplexe Zahlen immer Kartesisch gespeichert werden, nur die Anzeige ist betroffen. Dadurch bleibt die Genauigkeit in den Berechnungen weitgehendst erhalten.

Die meisten mathematischen Funktionen arbeiten mit komplexen Zahlen. Dadurch ist es nicht notwendig, dass Sie eine spezielle "komplex + " Funktion, zum hinzufügen von komplexen Zahlen verwenden, Sie können die gleiche + Funktion auf reelle oder Integer Zahlen anwenden.

Operationen mit Vektoren und Matrizen verwenden Objekte des Typs 3, reelle Arrays (Zahlenketten) , und falls benötigt Typ 4, **komplexe Arrays**. Objekte des Typs 2**, Strings** (Zeichenketten), sind einfache, mit der alpha-numerischen Tastatur erzeugte (zwischen Anführungszeichen gesetzte) Textzeilen.

Eine Liste ist nur eine Ansammlung von Objekten zwischen zwei geschwungenen Klammern, im RPN-Modus durch Leerschritte (die Leertaste ist als SPC beschriftet) oder im algebraischen Modus . durch Kommas getrennt. Listen, Objekte des Typs 5, sind besonders bei der Berechnung von Zahlensammlungen nützlich. So können z.B. können die Spalten einer Tabelle als Listen eingegeben werden. Falls gewünscht, kann die Tabelle auch als Matrix oder Array eingegeben werden.

Objekte des Typs 8 sind *Programme in User RPL* Sprache. Dies sind einfach Gruppen von Informationen, die zwischen den Symbolen << >> eingegeben werden.

Programmen zugeordnet sind auch Objekte des Typs 6 und 7, **globale** bzw. **lokale Namen**. Diese Namen oder Variablen werden zur Speicherung jeder Art von Objekten benutzt. Das Konzept von globalen oder lokalen Namen hängt vom Gültigkeits- oder dem Einflussbereich einer Variablen in einem Programm ab.

Ein **algebraisches Objekt** oder ganz einfach **Algebraik**, (Objekt des Typs 9), ist ein gültiger algebraischer Ausdruck zwischen zwei Apostrophe.

Ganze Dualzahlen , Objekte des Typs 10, werden in einigen computerwisschenschaftlichen Anwendungen verwendet.

Grafische Objekte , Objekte des Typs 11, speichern die vom Rechner erstellte Grafiken.

Gekennzeichnete Objekte, Objekte des Typs 12, werden als Ausgabe vieler Programme verwendet, um die Ergebnisse zu identifizieren. So z.B. bedeutet im gekennzeichneten Objekt Mean: 23,2, das Wort Mean: die Kennzeichnung zur Identifikation der Zahl 23,2 als Mittelwert einer Stichprobe.

Einheitenobjekte , Objekte des Typs 13, sind numerische Werte denen eine physikalische Einheit angehängt wurde.

Verzeichnisse , Objekte des Typs 15, sind Speicherstellen, welche bei der Organisation von Variablen, ähnlich den Verzeichnissen eines PC, behilflich sind.

Bibliotheken, Objekte des Typs 16, sind Programme im Speicher des Rechners, auf welche innerhalb eines Verzeichnisses (oder Unterverzeichnisses) in Ihrem Rechner zugegriffen werden kann. In deren Verwendung ähneln diese *built-in functions (eingebauten Funktionen)*, Objekte des Typs 18, und *built-in commands (eingebauten Befehlen)*, Objekte des Typs 19.

Ausdrücke im Display bearbeiten

In diesem Abschnitt werden Beispiele von Ausdrücken gezeigt, welche direkt im Display des Rechners (algebraische Veränderung oder RPN Stack) bearbeitet und verändert werden können.

Erstellen von arithmetischen Ausdrücken

Für dieses Beispiel wählen wir den algebraischen Modus und ein *Fix* (festes) Format mit 3 Dezimalstellen als Anzeige im Display. Wir geben nun den nachfolgenden arithmetischen Ausdruck ein:

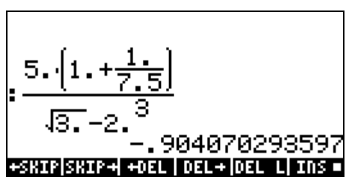
$$5.0 \cdot \frac{1.0 + \frac{1.0}{7.5}}{\sqrt{3.0} - 2.0^3}$$

Um diesen Ausdruck einzugeben verwenden Sie folgende Tastenfolge:

5 . × ← () 1 . + 1 . ÷ 7 . 5 ▶ ÷
← () √x 3 . − 2 . y^x 3

Der daraus resultierende Ausdruck sieht wie folgt aus: 5*(1/1+7.5)/(√3-2^3).

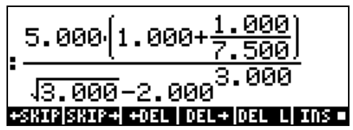
Drücken Sie die Taste ENTER, um den Ausdruck wie folgt anzuzeigen:



Beachten Sie dabei, dass, wenn Ihr CAS auf EXACT eingestellt ist (siehe Anhang C) und Sie Ihre Eingabe für Ganzzahlen über Integer-Werte machen, ist das Ergebnis eine symbolische Menge, z.B.

5 × ← () 1 + 1 ÷ 7 . 5 ▶ ÷
← () √x 3 − 2 y^x 3

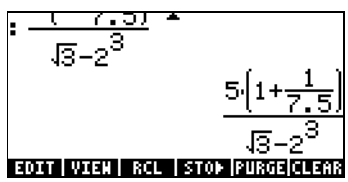
Bevor Sie ein Ergebnis erstellen, werden Sie darauf hingewiesen auf den Approx. mode (Näherungsmodus) umzustellen. Akzeptieren Sie die Änderung, um nachfolgendes Ergebnis zu erzielen (angezeigt im Fix decimal mode (Dezimalmodus) mit drei Nachkommastellen – siehe Kapitel 1):



In diesem Fall, wenn der Ausdruck direkt in den Stack eingegeben wird, wird der Rechner, sobald Sie die Taste ENTER drücken, versuchen einen Wert für den Ausdruck zu berechnen. Wird der Ausdruck aber in Anführungszeichen eingegeben, wird der Rechner den Ausdruck, genauso ausgeben, wie Sie diesen eingegeben haben.   Im nachfolgenden Beispiel, geben wir den gleichen Ausdruck, wie oben ein, aber verwenden dazu Anführungszeichen. In diesem Fall stellen wir auf den algebraischen Modus um, den CAS Modus auf Exact (entfernen das Häkchen bei _*Approx*), und stellen den Anzeige Modus auf *Textbook*.   Die dafür notwendigen Tastenanschläge sind wie folgt:

' 5 × ← () 1 + 1 ÷ 7 . 5 ▶ ÷
← () √x 3 − 2 y^x 3 ENTER

Das Ergebnis wird wie folgt angezeigt:



Um den Ausdruck auszuwerten können wir die Funktion EVAL, wie nachfolgend gezeigt, anwenden:

→ EVAL ← ANS

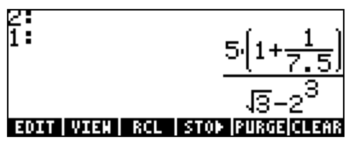
Wie im vorangegangenen Beispiel, werden Sie auch diesmal gefragt, ob Sie das CAS auf *Approx* umstellen wollen.   Sobald Sie dies getan haben, erhalten Sie das gleiche Ergebnis wie vorher.

Eine Alternative zu dem vorhin eingegebenen Ausdruck in Anführungszeichen auszuwerten, ist die Anwendung der Option [↱] →NUM . Um den Ausdruck aus dem existierenden Stack wieder herzustellen, benutzen Sie folgende Tastenfolge: [◀] [◀] [↱] →NUM

Wir geben nun den obigen Ausdruck ein, während der Rechner auf den RPN-Modus eingestellt ist. Wir setzen das CAS auf *Exact* und die Anzeige auf *Textbook*. Die Tastenfolge zur Eingabe des Ausdrucks zwischen Anführungszeichen ist die gleiche wie vorhin, d.h.

['] [5] [×] [↰] () [1] [+] [1] [÷] [7] [.] [5] [▶] [÷]
[↰] () [√x] [3] [−] [2] [yˣ] [3] [ENTER]

Dies ergibt die nachfolgende Ausgabe



Drücken Sie die Taste [ENTER] ein weiteres Mal, um zwei Kopien zur Auswertung des Ausdrucks im Stack zu behalten. Als erstes werten wir den Ausdruck mit der Funktion *EVAL* aus, anschließend verwenden wir die Funktion →*NUM*. Werten Sie zuerst den Ausdruck mit der Funktion EVAL aus. Dieser Ausdruck ist semi-symbolisch und zwar enthält er sowohl Gleitkomma-Komponenten als auch eine √3 im Ergebnis. Als nächstes, schalten wir die Stack Anordnung um und werten den Ausdruck mit der Funktion →NUM aus.

[▶]	Tauschen Sie die Stack Ebenen 1 und 2 (mit dem Befehl SWAP) aus
[↱] →NUM	Werten Sie diesen mit der Funktion →NUM aus

Dieses Ergebnis ist nun rein numerisch, so dass die Ergebnisse im Stack, obwohl Sie den gleichen Ausdruck darstellen, unterschiedlich aussehen. Um aber zu überprüfen, ob diese das gleiche Ergebnis liefern, subtrahieren wir beide Werte und berechnen diese mit der Funktion EVAL:

[−]	Subtrahieren Sie Ebene 1 von Ebene 2

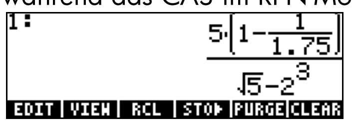
`→` `EVAL` Berechnen Sie anhand der Funktion EVAL

Das Ergebnis ist Null (0).

> **Anmerkung**: Vermeiden Sie es Integer- mit reellen Zahlen zu vermischen, um Konflikte in der Berechnung zu vermeiden. Für viele Anwendungen in der Physik und Technik, einschließlich numerischer Lösung von Gleichungen, Statistikanwendungen usw. funktioniert der APPROX Modus (siehe Anhang C) besser. Für mathematische Anwendungen, z.B. Calculus, Vektorenanalyse, Algebra usw. wird der EXACT Modus bevorzugt. Machen Sie sich mit beiden Methoden vertraut und lernen Sie, wie Sie aus einem in den anderen Modus, für unterschiedliche Berechnungen (siehe Anhang C) umschalten können.

Bearbeiten von arithmetischen Ausdrücken

Angenommen wir geben den folgenden Ausdruck, zwischen Anführungszeichen ein, während das CAS im RPN-Modus auf EXACT steht:



und nicht wie im nachfolgenden Ausdruck: $5\cdot\frac{1+\frac{1}{7.5}}{\sqrt{3}-2^3}$. Es wurde ein inkorrekter Ausdruck durch folgende Eingabe getätigt:

`→` `'` `5` `×` `←` `()` `1` `+` `1` `÷` `1` `.` `7` `5` `▶` `÷` `←` `()` `√x` `5` `−` `2` `yˣ` `3` `ENTER`

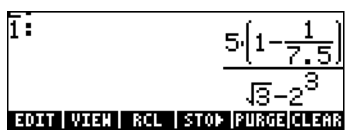
Um in den Zeileneditor zu gelangen drücken Sie `←` `▼`. Der Ausdruck sieht nun wie folgt aus:

Der Cursor zur Bearbeitung ist ein blinkender nach links gerichteter Pfeil, der sich über dem ersten Zeichen der zu bearbeitenden Zeile befindet. Da in diesem Fall die Bearbeitung im löschen einiger Zeichen und ersetzen dieser durch andere besteht, werden wir die Pfeiltasten ◀ ▶ dazu benutzen, den Cursor auf das zu verändernde Zeichen zu positionieren und anschließend die Löschtaste ◀ betätigen, um diese Zeichen zu entfernen.

Die nachfolgenden Tastenanschläge vervollständigen die Bearbeitung in diesem Fall:

- Drücken Sie die rechte Pfeiltaste ▶, bis sich der Cursor rechts des Dezimalkommas in der Zahl 1,75 befindet
- Drücken Sie die Löschtaste ◀ zweimal, um das Zeichen 1 zu löschen.
- Drücken Sie die rechte Pfeiltaste ▶, einmal, um den Cursor rechts der Zahl 7 zu positionieren
- Tippen Sie das Komma mit der Taste . ein
- Drücken Sie die rechte Pfeiltaste ▶, bis sich der Cursor rechts der Zahl 5 befindet
- Drücken Sie die Löschtaste ◀ einmal, um das Zeichen 5 zu löschen
- Geben Sie eine *3* mit der Taste 3 ein
- Drücken Sie ENTER um zum Stack zurückzukehren

Der bearbeitete Ausdruck befindet sich nun im Stack.



Die Bearbeitung einer Eingabezeile im algebraischen Modus erfolgt ganz genau wie im RPN-Modus. Sie können dieses Beispiel im algebraischen Modus wiederholen, um die Aussage zu überprüfen.

Erstellen von algebraischen Ausdrücken

Algebraische Ausdrücke beinhalten nicht nur Zahlen, sondern auch Namen von Variablen. Als Beispiel geben wir nachfolgenden algebraischen Ausdruck ein:

$$\frac{2L\sqrt{1+\frac{x}{R}}}{R+y}+2\frac{L}{b}$$

Wir stellen den algebraischen Operationsmodus am Rechner ein, CAS auf *Exact* und die Anzeige auf *Textbook*. Um diesen algebraischen Ausdruck einzugeben verwenden wir folgende Tastenfolge:

[→] [,] [2] [×] [ALPHA] [L] [×] [√x] [←] [()] [1] [+] [ALPHA] [←] [X] [÷] [ALPHA] [R] [▶]
[÷] [←] [()] [ALPHA] [R] [+] [ALPHA] [←] [Y] [▶] [+] [2] [×] [ALPHA] [L] [÷] [ALPHA] [←] [B]

Drücken Sie [ENTER], um nachfolgendes Ergebnis zu erhalten:

Genauso kann dieser Ausdruck im RPN-Modus eingegeben werden, was zum gleichen Ergebnis, wie im algebraischen Modus führt.

Bearbeiten von algebraischen Ausdrücken

Algebraische Ausdrücke werden mit dem Zeileneditor ähnlich wie arithmetische Ausdrücke bearbeitet (siehe Beispiel oben). Angenommen wir wollen den oben eingegebenen Ausdruck wie unten gezeigt verändern

$$\frac{2L\sqrt{1+\frac{x^2}{R}}}{R+x}+2\sqrt{\frac{L}{b}}$$

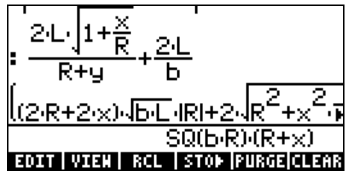
Um diesen algebraischen Ausdruck mit dem Zeileneditor zu bearbeiten benutzen wir [←] [▼]. Damit wird der Zeileneditor gestartet und der zu bearbeitende Ausdruck sieht wie folgt aus:

Der Cursor zur Bearbeitung ist ein blinkender nach links gerichteter Pfeil, der sich über dem ersten Zeichen der zu bearbeitenden Zeile befindet. Wie in einem früheren Beispiel zur Bearbeitung von Zeilen, werden wir die Pfeiltasten ◀ ▶, zur richtigen Positionierung des Cursors benutzen und anschließend die Löschtaste ◀, um entsprechende Zeichen zu löschen.

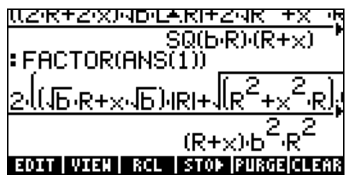
Die nachfolgenden Tastenanschläge vervollständigen die Bearbeitung in diesem Fall:

- Drücken Sie die rechte Pfeiltaste ▶, bis sich der Cursor rechts von x befindet
- Tippen Sie Y^x 2, um die Potenz 2 für x einzugeben
- Drücken Sie die rechte Pfeiltaste ▶, bis sich der Cursor rechts von y befindet
- Drücken Sie die Löschtaste ◀ einmal, um das Zeichen y zu löschen
- Drücken Sie ALPHA ↰ X um ein x einzugeben
- Drücken Sie die rechte Pfeiltaste ▶, viermal, um den Cursor rechts des * zu positionieren
- Drücken Sie √x um das Symbol Quadratwurzel einzugeben
- Drücken Sie ↰ (), um ein Klammernpaar einzugeben (gibt es immer paarweise)
- Drücken Sie die rechte Pfeiltaste ▶ einmal und anschließend die Löschtaste ◀ ebenfalls einmal, um die rechte Klammer des obigen Klammerpaares zu löschen.
- Drücken Sie die rechte Pfeiltaste ▶, viermal, um den Cursor rechts von *b* zu positionieren
- Geben Sie ↰ () ein, um ein zweites Klammerpaar zu öffnen
- Drücken Sie die Löschtaste ◀ einmal, um die linke Klammer des oben eingefügten Klammerpaares zu löschen.
- Drücken Sie die Taste ENTER , um zur Normalanzeige des Rechners zurückzukehren.

Nachfolgend das Ergebnis:



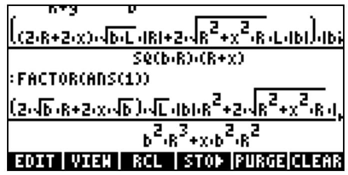
Beachten Sie, dass der Ausdruck erweitert wurde, um Faktoren, wie |R|, den absoluten Wert und SQ(b·R), die Quadratwurzel von b·R einzugeben. Um zu sehen, ob wir dieses Ergebnis vereinfachen können, benutzen wir FACTOR(ANS(1)) im ALG-Modus:



- Drücken Sie ← ▼, um den Zeileneditor ein weiteres Mal zu starten. Die Lösung lautet nun:
-

- Drücken Sie ENTER ein weiteres Mal, um zur Normalanzeige zurückzukehren.

Um den gesamten Ausdruck im Display zu sehen, ändern wir die Option auf _*Small Stack Disp* in der Eingabemaske *DISPLAY MODES* (siehe Kapitel 1). Nachdem Sie diese Änderung durchgeführt haben, sieht Ihre Anzeige wie folgt aus:



Anmerkung: Um griechische oder andere Buchstaben in algebraische Ausdrücke einzugeben, benutzen Sie das Menü CHARS. Dieses Menü wird mit der Tastenkombination ↱ CHARS gestartet. Einzelheiten finden Sie in Anhang D.

Erstellen von Ausdrücken mit Hilfe des EquationWriters (EQW) (Gleichungsschreibers)

Der EquationWriter ist ein wirklich starkes Werkzeug, welches Ihnen nicht nur erlaubt eine Gleichung einzugeben und anzusehen, sondern auch Änderungen vorzunehmen und zusätzliche Funktionen auf alle einzelnen Teile der Gleichung einzugeben und anzuwenden. Mit dem EquationWriter (EQW), können Sie komplexe mathematische Operation, direkt oder im Einzelschrittmodus durchführen, genauso wie Sie es mit Papier und Bleistift, z.B. bei der Lösung von Calculus Problemen tun würden.

Der EquationWriter wird durch drücken der Tastenkombination ↱↱ EQW gestartet (dritte Taste vierte Reihe von oben). Die Anzeige sieht wie folgt aus.

Die sechs Funktionstasten für den EquationWriter aktivieren die nachfolgenden Funktionen:

EDIT : damit kann der Anwender eine Eingabe mit dem Zeileneditor bearbeiten (siehe obige Beispiele)
CURS : markiert einen Ausdruck und fügt dieser einen grafischen Cursor hinzu

BIG : falls ausgewählt (die Auswahl wird durch ein Zeichen in der Beschriftung angezeigt) wird die Schriftgröße 8 im Editor verwendet (die größte vorhandene Schrift)

EVAL : damit können Sie einen im EquationWriter hervorgehobenen Ausdruck, symbolisch oder numerisch auswerten (ähnlich wie ↱ EVAL)

FACTO : damit können Sie einen im EquationWriter hervogehobenen Ausdruck faktorieren (falls eine Faktorierung möglich ist)

SIMP : lässt Sie einen Ausdruck in der EquationWriter Anzeige vereinfachen (sofern man diese gemäß algebraischen Regeln des CAS vereinfachen kann)

Drücken Sie die Taste NXT, erscheinen zwei weitere Funktionsmenüs, wie unten aufgezeigt:

Die sechs Funktionstasten für den EquationWriter starten die nachfolgenden Funktionen:

CMDS : ermöglicht Ihnen den Zugang zu den CAS Befehlen, welche in alphabetischer Reihenfolge aufgelistet sind. Dies bewährt sich bei der Eingabe von CAS Befehlen in einen im EquationWriter vorhandenen Ausdruck.

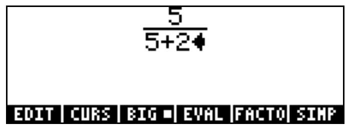
HELP : startet die CAS Hilfefunktion des Rechners, um Informationen und Beispiele zu CAS Befehlen zur Verfügung zu stellen.

Einige Beispiele in der Anwendung des EquationWriters werden unten aufgezeigt.

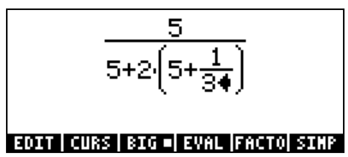
Erstellen von arithmetischen Ausdrücken

Die Eingabe von arithmetischen Ausdrücken in den EquationWriter ist ähnlich wie die Eingabe dieser in Anführungszeichen in den Stack. Der Hauptunterschied besteht darin, dass die in den EquationWriter eingegebenen Ausdrücke im "textbook" Stil (wie in einen Texteditor), anstelle von einer Eingabe Zeile für Zeile, geschrieben werden. Sobald Sie ein

Teilungszeichen (d.h. [÷]) in den EquationWriter eingeben, wird ein Bruch erzeugt und der Cursor in den Zähler gesetzt. Benutzen Sie die Pfeiltaste nach unten, um zum Nenner zu gelangen. Als Beispiel, versuchen Sie folgende Tastenfolge in den EquationWriter einzugeben:
[5][÷][5][+][2]
Das Ergebnis ist der Ausdruck

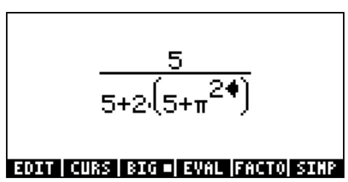


Der Cursor wird als ein nach links gerichteter Pfeil angezeigt. Der Cursor zeigt die momentane Position, in der eine Änderung vorgenommen werden kann, an. Die Eingabe eines Zeichens, Funktionsname oder Operation, wird an der Stelle erfolgen, an der sich der Cursor gerade befindet. So z.B., geben Sie an der Cursorposition, wie im Bild dargestellt, ein:
[×][←]() [5][+][1][÷][3]
Der bearbeitete Ausdruck sieht wie folgt aus:

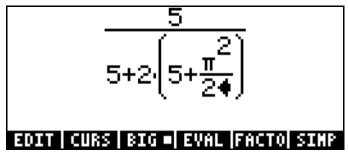


Angenommen Sie möchten den Ausdruck des Nenners in der Klammer ändern d.h. (5+1/3) mit (5+π^2/2) ersetzen. Als erstes benutzen wir die Löschtaste ([◄]) , löschen den Ausdruck 1/3 im Bruch und ersetzen diesen Teil mit π^2/2, wie folgt: [◄][◄][◄][←]π [y^x][2]

An dieser Stelle angekommen, sieht Ihre Anzeige wie folgt aus:



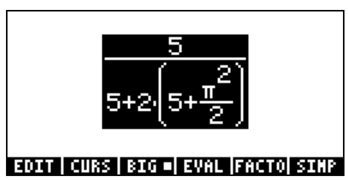
Um den Nenner 2 in den Ausdruck einzufügen, müssen wir den kompletten Ausdruck π^2 hervorheben (markieren). Dazu drücken wir die rechte Pfeiltaste (▶) einmal. An dieser Stelle fügen wir folgende Tastenfolge ein: ÷ 2
Der Ausdruck sieht nun wie folgt aus:



Angenommen Sie möchten den Bruch 1/3 zu diesem Ausdruck hinzufügen, d.h. Sie möchten folgenden Ausdruck eingeben:

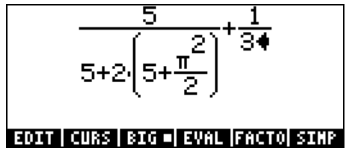
$$\frac{5}{5+2\cdot(5+\frac{\pi^2}{2})}+\frac{1}{3}$$

Als erstes, müssen wir den gesamten ersten Ausdruck, entweder mit der rechten Pfeiltaste (▶) oder der Taste nach oben (▲) hervorheben, und zwar den Vorgang so lange wiederholen bis der gesamte Ausdruck markiert ist, in diesem Fall, sieben Mal, wobei Ihre Anzeige dann wie folgt aussehen sollte:

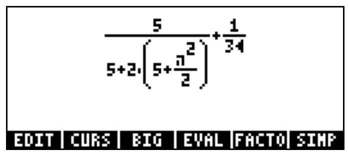


ANMERKUNG: Alternativ, von der Ursprungsposition des Cursors ausgehend (im Nenner rechts von der 2 im Ausdruck $\pi^2/2$), kann auch die Tastenkombination ↱ ▲, interpretiert als (↱ ⏶), verwendet werden.

Sobald der Ausdruck, wie oben gezeigt, hervorgehoben ist, tippen Sie folgende Tastenfolge ein, + 1 ÷ 3 ▶, um den Bruch 1/3 hinzuzufügen. Sie erhalten dann:



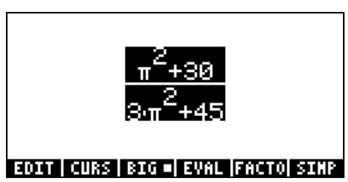
Den Ausdruck kleiner anzeigen
Um den Ausdruck in einer kleineren Schrift darzustellen (welche bei einem langen und weitverzweigten Ausdruck hilfreich sein kann), drücken Sie einfach die Funktionstaste BIG F3 . In diesem Fall sieht Ihre Anzeige wie folgt aus:



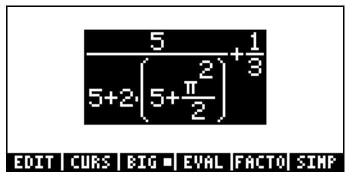
Um zur größeren Schriftart zurückzukehren, drücken Sie die Funktionstaste BIG F3 ein weiteres Mal.

Auswerten / berechnen des Ausdrucks
Um den Ausdruck (oder Teile des Ausdrucks) innerhalb des EquationWriters zu berechnen, markieren Sie den Teil, den Sie berechnen möchten und drücken die Funktionstaste EVAL F4 .

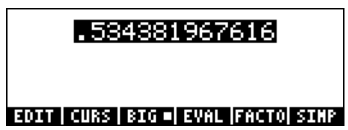
So z.B., um den gesamten Ausdruck in diesem Beispiel zu berechnen, markieren Sie zuerst den gesamten Ausdruck indem Sie ↱ ▲ drücken. Anschießend drücken Sie die Funktionstaste EVAL F4 . Befindet sich Ihr Rechner im Exakten CAS Modus (d.h. der *_Approx* CAS Modus ist nicht ausgewählt), erhalten Sie das nachfolgende symbolische Ergebnis:



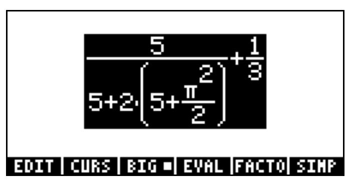
Möchten Sie den vorherigen, noch nicht berechneten Ausdruck, wiedergeben, benutzen Sie die Funktion UNDO, d.h. ↱ *UNDO* (die erste Taste in der dritten Reihe von oben). Der wiederhergestellte Ausdruck wird, wie vorhin markiert, angezeigt:



Wünschen Sie eine Gleitkommaberechnung (numerisch), benutzen Sie die Funktion →*NUM* (d.h. ↱ →*NUM*). Das Ergebnis sieht wie folgt aus:



Benutzen Sie Funktion UNDO (↱ *UNDO*) ein weiteres Mal, um den ursprünglichen Ausdruck wieder herzustellen:

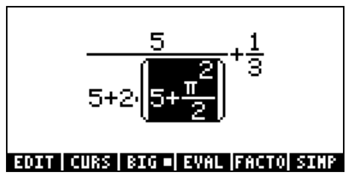


Berechnen eines Unterausdrucks

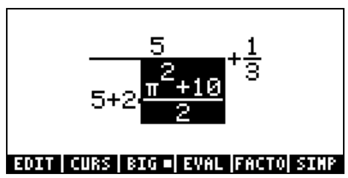
Angenommen aus obigem Ausdruck wollen Sie nur den in Klammer stehenden Ausdruck im Nenner des ersten Bruches berechnen. Benutzen Sie dazu die Pfeiltasten, um diesen bestimmten Unterausdruck auszuwählen. Nachfolgend eine Möglichkeit, wie Sie dies tun können:

▼ Hebt nur den ersten Bruck hervor
▼ Hebt den Zähler des ersten Bruches hervor
▶ Hebt den Nenner des ersten Bruches hervor
▼ Hebt das erste Glied im Nenner des ersten Bruches hervor

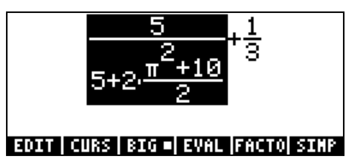
▶ Hebt das zweite Glied im Nenner des ersten Bruches hervor
▼ Hebt den ersten Faktor im zweiten Glied im Nenner des ersten Bruches hervor
▶ Hebt den in Klammer stehenden Ausdruck im Nenner des ersten Bruches hervor



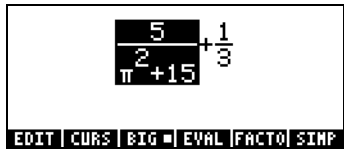
Da dies der Unterausdruck ist, den wir berechnen möchten, können wir an dieser Stelle die Funktionstaste EVAL F4 drücken, welche uns nachfolgendes Ergebnis anzeigt:



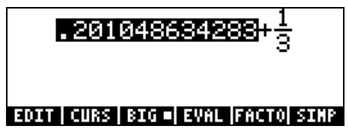
Eine weitere symbolische Berechnung. Angenommen an dieser Stelle möchten wir nur den Bruch auf der linken Seite berechnen. Drücken Sie die Pfeiltaste (▲) dreimal, um diesen Bruch auszuwählen, was dann wie folgt aussieht:



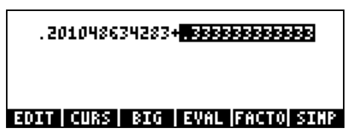
Anschießend drücken Sie die Funktionstaste EVAL F4 um den nachfolgenden Ausdruck zu erhalten:



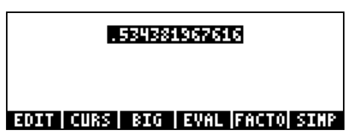
Versuchen wir es an dieser Stelle nun mit einer numerischen Berechnung dieses Gliedes. Verwenden Sie dazu ↱ →NUM , um nachfolgendes Ergebnis zu erhalten:



Heben wir nun den Bruch auf der rechten Seite hervor, um auch für dieses Glied eine numerische Berechnung zu erhalten und lassen wir die Summe der beiden Dezimalwerte in kleiner Schrift, unter Verwendung von: ▶ ↱ →NUM F3 anzeigen, dann erhalten wir:

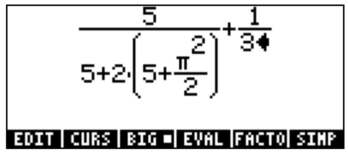


Um den Ausdruck hervorzuheben und im EquationWriter zu berechnen, benutzen wir ▲ F4 , wodurch wir folgendes Ergebnis erzielen:

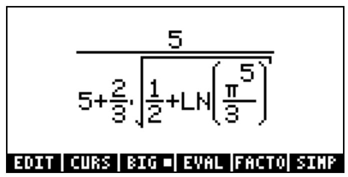


Bearbeiten von arithmetischen Ausdrücken

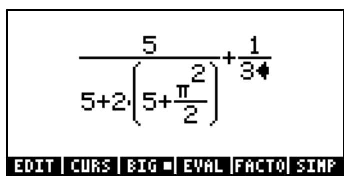
Nachfolgend zeigen wir einige Bearbeitungsmerkmale im EquationWriter als Beispiel. Wir beginnen, indem wir den im vorherigen Beispiel verwendeten Ausdruck eingeben:



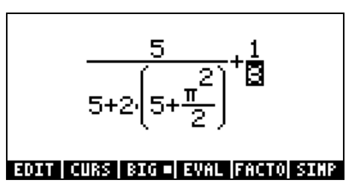
Dann verwenden wir die Bearbeitungsmerkmale des EquationWriters, um diesen wie nachfolgend umzuwandeln:



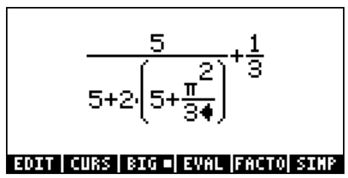
In vorangegangenen Übungen, haben wir die Pfeiltasten zur Markierung von Unterausdrücken für Berechnungen, verwendet. In diesem Fall benutzen wir sie dazu einen bestimmten Bearbeitungscursor auszuwählen. Nachdem Sie die Eingabe des ursprünglichen Ausdrucks vervollständigt haben, wird der Eingabecursor (ein nach links gerichteter Pfeil) rechts von der 3 im Nenner des zweiten Bruches, wie unten gezeigt, stehen:



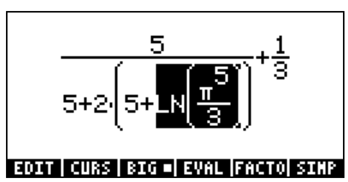
Drücken Sie die Pfeiltaste (▼), um zum reinen Bearbeitungscursor zu gelangen. Die Anzeige sieht nun wie folgt aus:



Mit der linken Pfeiltaste ([◄]) können Sie den Cursor im allgemeinen nach links bewegen, dieser hält aber bei jeder einzelnen Komponente des Ausdrucks. Nehmen wir z.B. an, dass wir als erstes den Ausdruck $\pi^2/2$ in den Ausdruck $LN(\pi^5/3)$ umwandeln wollen. Mit der aktivierten reinen Cursortaste, wie oben angezeigt, drücken Sie die Pfeiltaste [◄] zweimal, um die 2 im Nenner von $\pi^2/2$ hervorzuheben. Drücken Sie als nächstes die Löschtaste [⇦] einmal, um den Cursor in einen Einfügecursor umzuwandeln. Drücken Sie die Taste [⇦] ein zweites Mal, um die 2 zu löschen, um dann [3] die 3 einzugeben. An dieser Stelle sieht Ihre Anzeige wie folgt aus:



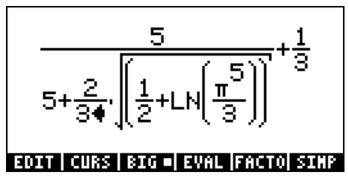
Als nächstes drücken Sie die Pfeiltaste [▼] , um mit dem reinen Bearbeitungscursor die 3 im Nenner von $\pi^2/3$ hervorzuheben. Drücken Sie die Pfeiltaste [◄] ein weiteres Mal, um den Exponenten im Ausdruck $\pi^2/3$ hervorzuheben. Drücken Sie als nächstes die Löschtaste [⇦] einmal, um den Cursor in einen Einfügecursor umzuwandeln. Drücken Sie die Taste [⇦] ein zweites Mal, um die 2 zu löschen, um dann die 5 über die Taste [5] einzugeben. Drücken Sie die Pfeiltaste [▲] dreimal, um den Ausdruck $\pi^2/3$ hervorzuheben. Tippen Sie anschließend [↱] LN , um die Funktion *LN* auf diesen Ausdruck anzuwenden. Die Anzeige sieht nun wie folgt aus:



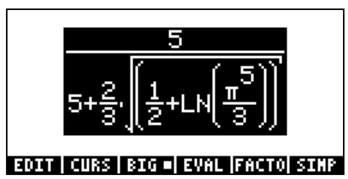
Als nächstes werden wir die 5 innerhalb der Klammern in ½ ändern, indem wir nachfolgende Tastenanschläge benutzen:

[◄] [⇦] [⇦] [1] [÷] [2]

Dann markieren wir den gesamten Ausdruck in der Klammer und fügen das Quadratwurzelzeichen wie folgt ein: ▲▲▲▲√x
Als Nächstes konvertieren wir die 2 vor der Klammer des Nenners in 2/3 wie folgt: ◄ ⌫ ⌫ 2 ÷ 3
An dieser Stelle sieht Ihre Anzeige wie folgt aus:



Der letzte Schritt ist 1/3 rechts vom Ausdruck zu entfernen. Dies wird wie folgt erreicht: ▲▲▲▲▲► ⌫ ⌫ ⌫ ⌫ ⌫
Die endgültige Version sieht nun so aus:



Zusammenfassend gesagt, um einen Ausdruck im EquationWriter zu bearbeiten sollten Sie die Pfeiltasten ◄►▲▼ benutzen, um den Ausdruck, auf welchen Sie eine Funktion anwenden wollen, hervorzuheben (z.B. die Funktion LN und Quadratwurzel im obigen Beispiel). Drücken Sie wiederholt die Pfeiltaste ▼, an jeder x-beliebigen Stelle, um zum reinen Bearbeitungscursor zu gelangen. In diesem Modus verwenden Sie dann die Pfeiltasten ◄►, um sich von einem Glied zum nächsten im Ausdruck zu bewegen. An einem der Punkte, die Sie bearbeiten möchten, angekommen, benutzen Sie die Löschtaste ⌫, um den Einfügecursor zu wählen und fahren Sie mit der Bearbeitung des Ausdrucks fort.

Erstellen von algebraischen Ausdrücken

Ein algebraischer Ausdruck ähnelt einem arithmetischen, mit Ausnahme, dass in den algebraischen auch noch englische oder griechische Buchstaben eingefügt werden können. Der Vorgang beim Erstellen eines algebraischen Ausdrucks ist der gleiche wie beim Erstellen eines arithmetischen Ausdrucks, mit Ausnahme, dass eine alphabetische Tastatur zur Verfügung steht.

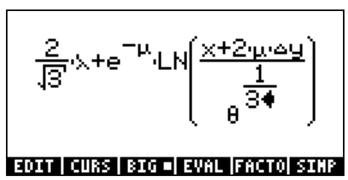
Nehmen wir nachfolgendes Beispiel, um die Anwendung des EquationWriters bei der Eingabe eines algebraischen Ausdrucks zu veranschaulichen. Angenommen wir möchten folgenden Ausdruck eingeben.

$$\frac{2}{\sqrt{3}}\lambda + e^{-\mu} \cdot LN\left(\frac{x + 2\mu \cdot \Delta y}{\theta^{1/3}}\right)$$

Verwenden Sie dazu folgende Tastenfolge:

[2] [÷] [√x] [3] [▶] [▶] [×] [ALPHA] [↱] [N] [+] [↰] e^x [+/-] [ALPHA] [↱] [M]
[▶] [▶] [×] [↱] LN [ALPHA] [↰] [X] [+] [2] [×] [ALPHA] [↱] [M] [×] [ALPHA] [↱] [C]
[ALPHA] [↰] [Y] [▲] [▲] [▲] [÷] [ALPHA] [↱] [T] [y^x] [1] [÷] [3]

Dies ergibt die nachfolgende Ausgabe:



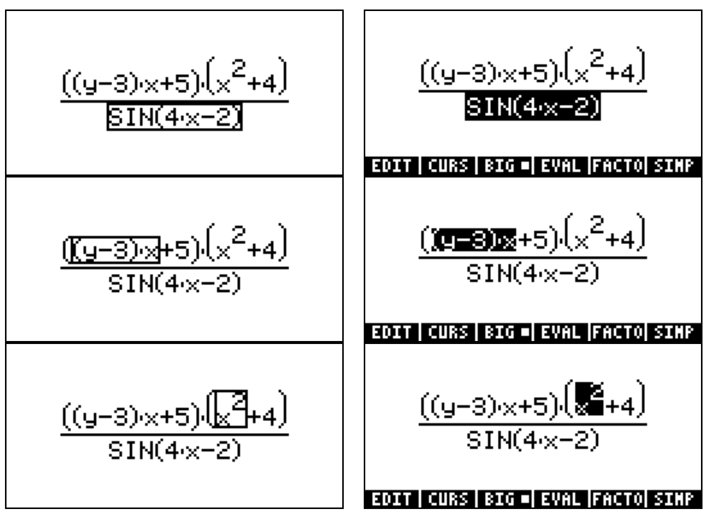
In diesem Beispiel haben wir mehrere englische Kleinbuchstaben verwendet, z.B. ., x ([ALPHA] [↰] [X]), mehrere griechische Buchstaben, z.B. λ ([ALPHA] [↱] [N]), aber auch eine Kombination aus englischen und griechischen Buchstaben, Δy ([ALPHA] [↱] [C]
[ALPHA] [↰] [Y]). Vergessen Sie dabei nicht, um einen englischen Kleinbuchstaben eingeben zu können, benötigen Sie die Kombination: [ALPHA] [↰] gefolgt vom Buchstaben, den Sie eingeben möchten. Sie können auch Sonderzeichen mit Hilfe des Menüs CHARS ([↱] CHARS) eingeben, wenn Sie sich die Tastenkombinationen für diese nicht alle merken möchten. Eine Auflistung häufig verwendeter [ALPHA] [↱] Tastenkombinationen wurde in einem vorangegangenen Abschnitt aufgeführt.

Der Ausdrucksbaum

Der Ausdrucksbaum ist ein Diagramm, das anzeigt, wie der EquationWriter einen Ausdruck darstellt (interpretiert). Für ein ausführliches Beispiel siehe Anhang E.

Die Funktion CURS

Die Funktion CURS (CURS) im EquationWriter (Taste F2) konvertiert die Anzeige in eine grafische Anzeige und erzeugt einen grafischen Cursor, der über die Pfeiltasten ◀ ▶ ▲ ▼ gesteuert werden kann, um Unterausdrücke auszuwählen. Der mit CURS ausgewählte Unterausdruck, wird in der grafischen Anzeige als gerahmt dargestellt. Nachdem Sie einen Unterausdruck ausgewählt haben, können Sie die Taste ENTER drücken, um den ausgewählten Unterausdruck als hervorgehoben im EquationWriter anzuzeigen. Durch drücken der Taste ENTER zeigen die nachfolgenden Abbildungen unterschiedliche ausgewählte Unterausdrücke und die entsprechende EquationWriter Anzeige.

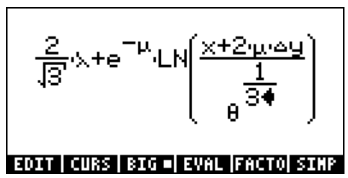


Bearbeiten von algebraischen Ausdrücken

In der Bearbeitung algebraischer Ausdrücke gelten die gleichen Regeln wie in der Bearbeitung von algebraischen Gleichungen. Und das wäre:

- Benutzen Sie die Pfeiltasten ( ) um den Ausdruck zu markieren
- Drücken Sie wiederholt den Pfeil ( ), um zum reinen Bearbeitungscursor zu gelangen. In diesem Modus verwenden Sie dann die Pfeiltasten ( ), um sich von einem Glied zum nächsten im Ausdruck zu bewegen.
- Benutzen Sie die Löschtaste ( ) an einem Bearbeitungspunkt, um den Eingabecursor zu wählen und bearbeiten Sie den Ausdruck.

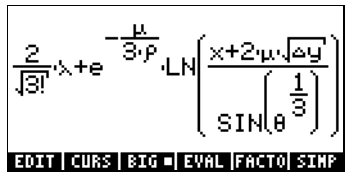
Um den reinen Bearbeitungscursor in Aktion zu sehen, beginnen wir mit dem algebraischen Ausdruck, welchen wir im obigen Beispiel eingefügt haben:



Drücken Sie den Pfeil , da wo er sich gerade befindet, um zum reinen Bearbeitungscursor zu gelangen. Die 3 im Exponenten θ wird hervorgehoben. Benutzen Sie die Pfeiltaste , um sich von einem Element des Ausdrucks zum anderen zu bewegen. Die Auswahlreihenfolge des reinen Bearbeitungscursors in unserem Beispiel ist die nachfolgende (drücken Sie wiederholt die linke Pfeiltaste):

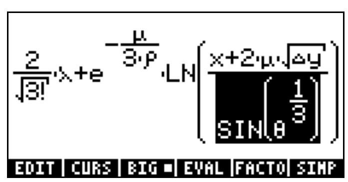
1. Die 1 im Exponenten von 1/3
2. θ
3. Δy
4. μ
5. 2
6. x
7. μ in der Exponentialfunktion
8. λ
9. 3 in der √3
10. zwei im Bruch 2/√3

An dieser Stelle können wir den reinen Bearbeitungscursor in einen Einfügecursor ändern, indem wir die Löschtaste drücken. Benutzen wir nun diese beiden Cursor (den reinen Bearbeitungscursor und den Einfügecursor), um den aktuellen Ausdruck wie folgt zu ändern:



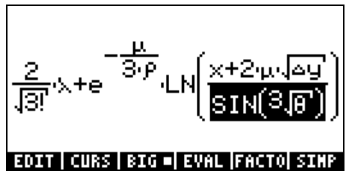
Haben Sie die Übung von vorhin gleich durchgeführt, sollten Sie jetzt den nachfolgenden reinen Bearbeitungscursor auf der Zahl 2 im ersten Faktor des Ausdrucks haben. Führen Sie nachfolgenden Tastenanschläge durch, um den Ausdruck zu bearbeiten:

▶ ALPHA ↱ 2	Trägt die Faktorielle für die 3 in der Quadratwurzel ein (das Eintragen der Faktoriellen ändert den Cursor in einen Auswahlcursor)
▼ ▼ ▶ ▶	Wählt das μ in der Exponentialfunktion
÷ 3 × ALPHA ↱ F	Ändert das Argument der Exponentialfunktion
▶ ▶ ▶ ▶	Wählt *Δy* aus
√x	Platziert ein Quadratwurzel Symbol über *Δy* (auch dieser Vorgang ändert den Cursor in den Auswahlcursor)
▼ ▼ ▶ ▲ ▲ SIN	Wählen Sie $\theta^{1/3}$ und tragen Sie die Funktion SIN ein Die Anzeige sieht wie folgt aus.

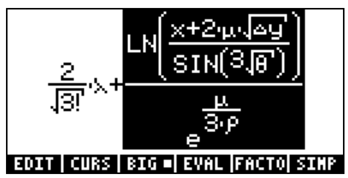


Berechnen eines Unterausdrucks

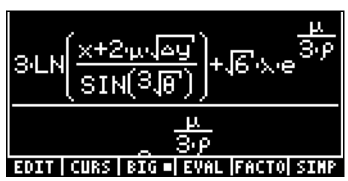
Da wir den Unterausdruck $\mathrm{SIN}\left(\theta^{1/3}\right)$ bereits hervorgehoben haben, drücken wir nun die Funktionstaste EVAL F4, um diesen Unterausdruck zu berechnen. Die Lösung lautet:



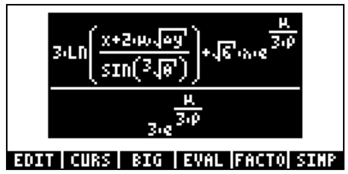
Einige algebraische Ausdrücke können nicht weiter vereinfacht warden. Versuchen Sie folgende Tastenaschläge: ▲ F4 . Sie werden feststellen, nur das komplette Argument der Funktion *LN* hervorgehoben wird, sonst passiert nichts weiter. Dies ist der Fall, weil der Ausdruck, nach den Regeln des CAS, nicht weiter berechnet (oder vereinfacht) werden kann. Versuchen Sie die Tastenfolge: ▲ F4 erneut, gibt es immer noch keine Änderungen im Ausdruck. Ein weiterer Versuch mit dieser Tastenfolge ▲ F4 hingegen, ändert den Ausdruck, wie folgt:



Betätigen Sie diese Tastenfolge ▲ F4 ein weiteres Mal, erscheinen weitere Änderungen:



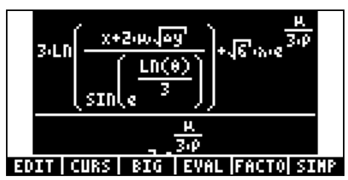
Dieser Ausdruck passt nicht mehr in den Anzeigebildschirm des EquationWriters. Der gesamte Ausdruck lässt sich aber mit einer kleineren Schrift ansehen. Drücken Sie die Funktionstaste BIG F3 , um nachfolgendes zu erhalten:



Auch mit der größeren Schrift ist es möglich sich durch den gesamten Ausdruck mit dem reinen Bearbeitungscursor zu bewegen. Versuchen Sie folgende Tastenfolge: F3 ▼▼▼▼, um den reinen Bearbeitungscursor über den Faktor 3 im ersten Glied des Zählers zu bewegen. Drücken Sie die Pfeiltaste ▶, um sich durch den Ausdruck zu bewegen.

Einen Ausdruck vereinfachen

Drücken Sie die Funktionstaste BIG F3, um die Anzeige, wie in der vorangegangenen Abbildung (siehe oben) zu erhalten. Drücken Sie nun die Funktionstaste SIMP F3, um zu sehen, ob dieser Ausdruck, wie er sich im EquationWriter befindet, noch weiter vereinfacht werden kann. Das Ergebnis ist die nachfolgende Anzeige:



Die Anzeige zeigt das Argument der Funktion SIN, und zwar, $\sqrt[3]{\theta}$ umgewandelt in $e^{\frac{LN(\theta)}{3}}$. Dies mag zwar nicht wie eine Vereinfachung aussehen, aber es ist eine im Sinne von, dass die Quadratwurzelfunktion mit der inversen Funktion exp-LN ersetzt wurde.

Faktorieren eines Ausdrucks

In diesem Beispiel werden wir versuchen einen Polynomausdruck zu faktorieren. Um mit dem vorangegangenen Beispiel weiterzumachen, drücken Sie die Taste [ENTER]. Starten Sie dann den EquationWriter erneut durch drücken der Tasten [↱] EQW . Geben Sie folgende Gleichung ein:

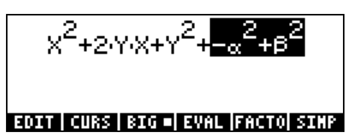
[X][Y^x][2][▶][+][2][×][X][×][ALPHA][Y][+][ALPHA][Y] [Y^x][2][▶][−]
[ALPHA][↱][A] [Y^x][2][▶][▶][+][ALPHA][↱][B] [Y^x][2]

dies ergibt dann:

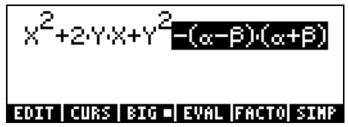
Wählen wir nun die ersten 3 Glieder des Ausdruckes aus und versuchen wir diesen Unterausdruck zu faktorieren: [↱][▲][▼][↱][▶][↱][▶] . Dies ergibt:

Drücken Sie die Funktionstaste [FACTO] , um nachfolgendes zu erhalten:

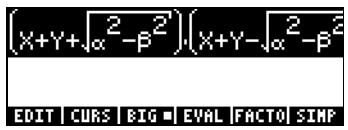
Um zum ursprünglichen Ausdruck zurückzukehren drücken Sie [↱] UNDO . Als nächstes geben Sie folgende Tastefolge ein:
[▼][▼][▼][▶][▶][▶][▶][▶][▶][▶][▲][▲][▲][↱][▶] ,um die letzten beiden Glieder im Ausdruck zu markieren, d.h.,



Drücken Sie die Funktionstaste FACTO , um nachfolgendes zu erhalten:



Um zum ursprünglichen Ausdruck zurückzukehren drücken Sie ↱ UNDO . Markieren wir nun den gesamten Ausdruck, indem wir die Pfeiltaste (▲) einmal drücken. Drücken Sie die Funktionstaste FACTO , um nachfolgendes zu erhalten:



Um zum ursprünglichen Ausdruck zurückzukehren drücken Sie ↱ UNDO .

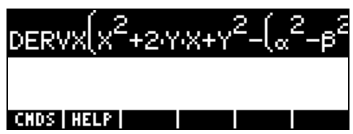
Anmerkung: Drücken Sie die Funktionstaste EVAL oder SIMP , wenn der gesamte Ausdruck hervorgehoben ist, erhalten Sie nachfolgende Vereinfachung des Ausdruckes:

Verwenden der Menütaste CMDS

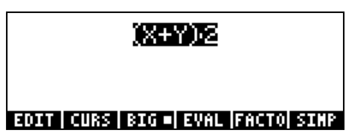
Verwenden wir den Polynomenausdruck aus vorangegangenem Beispiel und drücken die Taste NXT, um die Funktionstasten des Menüs CMDS und HELP anzuzeigen. Diese beiden Befehle gehören zum zweiten Teil des vorhandenen Softwaremenüs im EquationWriter. Versuchen wir nun ein Beispiel in dem die Funktionstaste CMDS zur Anwendung kommt: Drücken Sie die Funktionstaste CMDS , um eine Auflistung der CAS Befehle zu erhalten:

Als nächstes wählen Sie den Befehl DERVX (die Ableitungsfunktion in Bezug auf die Variable X, die aktuelle unabhängige Variable im CAS) wie folgt: ALPHA D ▼ ▼ ▼ . Nun wird der Befehl DERVX ausgewählt:

Drücken Sie die Funktionstaste OK (F6), um nachfolgendes zu erhalten:



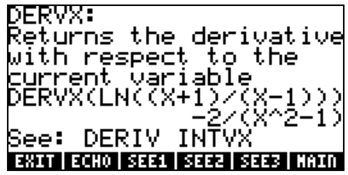
Als Nächstes, drücken Sie die Taste NXT, um zum ursprünglichen EquationWriter Menü zurückzukehren und anschließend die Funktionstaste EVAL (F4), um diese Ableitungsfunktion zu berechnen. Die Lösung lautet:



Verwenden des Menüs HELP (Hilfe)

Drücken Sie die Taste NXT, um die Funktionstasten CMDS und HELP anzuzeigen. Drücken Sie die Funktionstaste HELP, um eine Auflistung der CAS Befehle zu erhalten: Drücken Sie dann ALPHA D ▼ ▼ ▼, um den

Befehl DERVX auszuwählen. Drücken Sie die Funktionstaste OK (F6), um Informationen zum Befehl DERVX zu erhalten:



Eine genaue Erklärung zur Verwendung der Hilfefunktion für das CAS finden Sie in Kapitel 1. Um zum EquationWriter zurückzukehren, drücken Sie die Funktionstaste EXIT. Drücken Sie ENTER, um den EquationWriter zu verlassen.

Verwenden der Funktionen BEGIN, END, COPY, CUT und PASTE

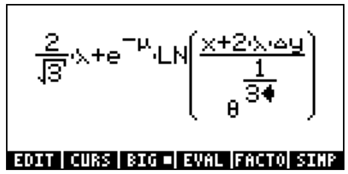
Um die Bearbeitung zu vereinfachen, egal ob Sie sich im EquationWrier oder im Stack befinden, werden im Rechner fünf Bearbeitungsfunktionen zur Verfügung gestellt BEGIN, END, COPY, CUT und PASTE, welche in Kombination mit der rechten Shift-Taste (↱) und den entsprechenden Tasten (2,1), (2,2), (3,1), (3,2) und (3,3) gestartet werden. Dies sind die ganz links angeordneten Tasten der Reihen 2 und 3. Die Aktion dieser Bearbeitungsfunktionen sind wie folgt:

BEGIN: markiert den Anfang einer Zeichenkette zur Bearbeitung
END: markiert das Ende einer Zeichenkette zur Bearbeitung
COPY: kopiert die Zeichenkette, die mit BEGIN und END ausgewählt wurde
CUT: schneidet die Zeichenkette, die mit BEGIN und END ausgewählt wurde, aus
PASTE: kopiert, die vorher kopierte oder ausgeschnittene Zeichenkette, an die aktuelle Cursorposition

Um ein Beispiel zu sehen, starten wir den EquationWriter und geben den nachfolgenden Ausdruck ein (wurde in einem vorangegangenen Beispiel benutzt):

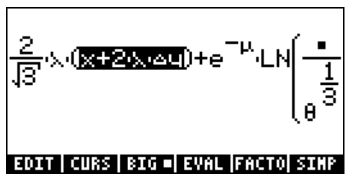
2 ÷ √x 3 ▶ ▶ × ALPHA ↱ M + ↰ eˣ +/- ALPHA ↱ M
▶ ▶ × ↱ LN ALPHA ↰ X + 2 × ALPHA ↱ M × ALPHA ↱ C
ALPHA ↰ Y ▲ ▲ ▲ ÷ ALPHA ↱ T yˣ 1 ÷ 3

Der ursprüngliche Ausdruck sieht wie folgt aus.

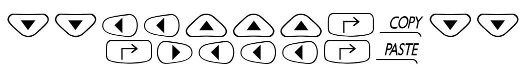


Wir wollen nun den Unterausdruck $x+2\cdot\lambda\cdot\Delta y$ aus dem Argument der Funktion *LN* entfernen und nach rechts von λ im ersten Glied verschieben. Eine Möglichkeit dies zu tun: 

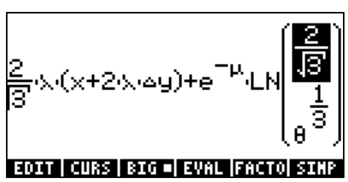
Der veränderte Ausdruck sieht wie folgt aus:



Anschließend kopieren wir den Bruch $2/\sqrt{3}$ aus dem Ausdruck ganz links und platzieren diesen in den Zähler des Argumentes der Funktion LN. Versuchen Sie folgende Tastenaschläge:



Die Anzeige wie folgt aus:

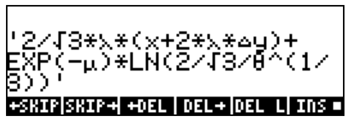


Wenn Sie im EquationWriter arbeiten, benötigen Sie die Funktionen BEGIN und END nicht, da wir Zeichenketten mit den Pfeiltasten auswählen können. Die Funktionen BEGIN und END werden hauptsächlich dann benötigt, wenn wir uns im Zeileneditor befinden. Wählen wir z.B. den Ausdruck $x+2\cdot\lambda\cdot\Delta y$

aus diesem Ausdruck, aber unter Verwendung des Zeileneditor innerhalb des EquationWriters, wie folgt:

↱ ▲ F1

Die Anzeige des Zeileneditors sieht wie folgt aus:



Um den gewünschten Unterausdruck auszuwählen, verwenden Sie:

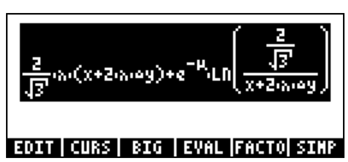
▶ ▶ ▶ ▶ ▶ ▶ ▶ ▶ ↱ BEGIN
▶ ▶ ▶ ▶ ▶ ▶ ▶ ▶ ▶ ▶ ↱ END

In der Anzeige unten ist der gewünschte Unterausdruck hervorgehoben:

Wir können nun diesen Ausdruck kopieren und im Nenner des Argumentes der Funktion LN platzieren, wie folgt: ↱ COPY ▶ ▶ … (27 Mal) … ▶
◀ ◀ … (9 Mal) … ◀ ↱ PASTE

Die Anzeige des Zeileneditors sieht nun wie folgt aus:

Drücken Sie ENTER erhalten Sie den Ausdruck im EquationWriter (in kleiner Schrift, drücken Sie die Funktionstaste BIG F3):



Drücken Sie [ENTER], um den EquationWriter zu verlassen.

Erstellen und bearbeiten von Summen, Ableitungsfunktionen und Integrale

Summen, Ableitungsfunktionen und Integrale werden im allgemeinen bei Calculus-, Wahrscheinlichkeits- und Statistik-Anwendungen eingesetzt. In diesem Abschnitt zeigen wir einige Beispiele solcher Operationen, erstellt mit dem EquationWriter.

Summen

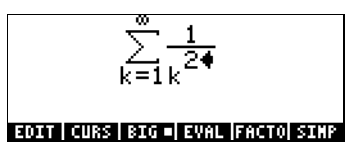
Wir benutzen den EquationWriter, um folgende Summe einzugeben:

$$\sum_{k=1}^{\infty} \frac{1}{k^2}$$

Drücken Sie [→] EQW , um den EquationWriter zu starten. Anschließend drücken Sie dann [→] Σ , um das Summenzeichen einzugeben. Beachten Sie, dass die Anzeige des EquationWriters, Eingabefelder für den Index der Summe, aber auch für die zu summierende Menge, zur Verfügung stellt. Um diese Eingabefelder auszufüllen, benutzen Sie die Tastefolge:

[ALPHA][←][K][▶][1][▶][←] ∞ [▶][1][÷][ALPHA][←][K][Y^x][2]

Die daraus resultierende Anzeige sieht wie folgt aus:



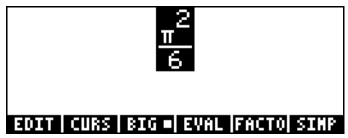
Um den entsprechenden Ausdruck im Zeileneditor anzuzeigen, drücken Sie [→][▲] und die Funktionstaste [F1] , um folgende Ansicht zu erhalten:

Dieser Ausdruck zeigt eine allgemeine Form einer Summe, die direkt in den Stack oder Zeileneditor eingegeben wurde:

Σ(*index = starting_value, ending_value, summation expression*)

Σ(*Index = Anfangswert, Endwert, Summenausdruck*)

Drücken Sie ENTER, um zum EquationWriter zurückzukehren. Die so entstandene Anzeige ist nicht die Summe, die wir eingegeben haben, sonderen deren symbolischer Wert, und zwar:



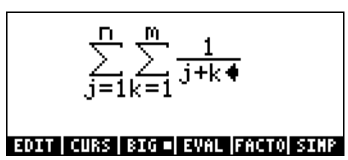
Um die Summe wieder herzustellen, verwenden Sie → UNDO . Um die Summe neu zu berechnen, verwenden Sie die Funktionstaste F4 . Diese zeigt erneut, dass

$$\sum_{k=1}^{\infty} \frac{1}{k^2} = \frac{\pi^2}{6} .$$

Sie können den EquationWriter benutzen, um zu beweisen, dass

$$\sum_{k=1}^{\infty} \frac{1}{k} = +\infty .$$

diese Summe (eine unendliche Reihe darstellend) divergent sei.
Auch doppelte Summen sind möglich, so z.B.:



Ableitungsfunktionen

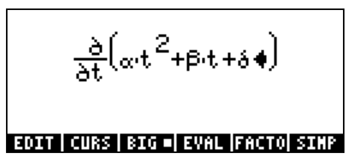
Wir benutzen den EquationWriter, um folgende Ableitungsfunktion einzugeben:

$$\frac{d}{dt}(\alpha \cdot t^2 + \beta \cdot t + \delta)$$

Drücken Sie [→] EQW , um den EquationWriter zu starten. Drücken Sie anschließend [→] ∂ , um in das (partielle) Ableitungsfunktionszeichen zu gelangen. Beachten Sie, dass das Zeichen, wenn in den EquationWriter eingegeben, eine Eingabemöglichkeit für den abzuleitenden Ausdruck und die Ableitungsvariable zur Verfügung stellt. Um diese Eingabefelder auszufüllen, benutzen Sie die Tastefolge:

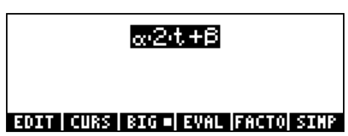
[ALPHA] [←] [T] [▶] [ALPHA] [→] [A] [×] [ALPHA] [←] [T] [Y^x] [2]
[▶] [▶] [+] [ALPHA] [→] [B] [×] [ALPHA] [←] [T] [+] [ALPHA] [→] [D]

Die Anzeige sieht wie folgt aus.



Um den entsprechenden Ausdruck im Zeileneditor anzuzeigen, drücken Sie [→] [▲] und die Funktionstaste [F1] , um folgende Ansicht zu erhalten:

Daraus ersehen wir, dass der allgemeine Ausdruck einer Ableitungsfunktion im Zeileneditor oder Stack wie folgt lautet:
∂*Variable*(*Funktion von Variablen*)
Drücken Sie [ENTER], um zum EquationWriter zurückzukehren. Die so entstandene Anzeige ist nicht die Ableitungsfunktion, die wir eingegeben haben, sonderen deren symbolischer Wert, und zwar:

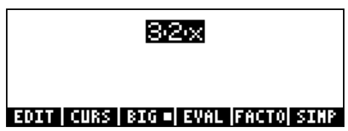


Um die Ableitungsfunktion wieder herzustellen, verwenden Sie ↱ UNDO . Um die Ableitungsfunktion neu zu berechnen, verwenden Sie die Funktionstaste F4 . Diese zeigt erneut, dass

$$\frac{d}{dt}(\alpha \cdot t^2 - \beta \cdot t + \delta) = 2\alpha \cdot t + \beta \ .$$

Auch eine Ableitung aus einer Ableitung ist möglich, so z.B.:

welche ausgewertet



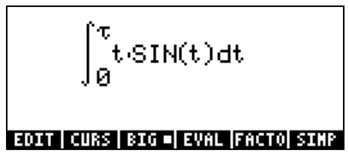
ergibt.

Anmerkung: Für eine Teilableitung ist die Notation $\frac{\partial}{\partial x}(\ )$ so korrekt. Die korrekte Notation für eine Gesamtableitung (d.h. eine Ableitung einer Variablen) lautet $\frac{d}{dx}(\ )$. Der Rechner macht aber keinen Unterschied zwischen Teil- und Gesamtableitungen.

Bestimmte Integrale

Wir benutzen den EquationWriter, um folgende bestimmte Integrale $\int_0^\tau t \cdot \sin(t) \cdot dt$ einzugeben. Drücken Sie ↱ EQW , um den EquationWriter zu starten. Drücken Sie anschließend ↱ ∫ , um in das Integralzeichen zu

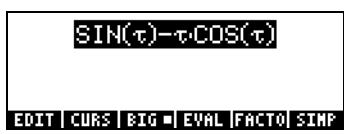
gelangen. Beachten Sie, dass das Zeichen, wenn in den EquationWriter eingegeben, eine Eingabemöglichkeit für die Grenzwerte der Integrale, den Integranden und die Integrationsvariable zur Verfügung stellt. Um diese Eingabefelder auszufüllen, benutzen Sie die Tastefolge:
[0] [▶] [ALPHA] [↱] [U] [▶] [ALPHA] [↰]
[T] [×] [SIN] [ALPHA] [↰] [T] [▶] [ALPHA] [↰] [T] . Die Anzeige sieht wie folgt aus.



Um den entsprechenden Ausdruck im Zeileneditor anzuzeigen, drücken Sie [▲][▲] und die Funktionstaste [F1] , um folgende Ansicht zu erhalten:

Daraus ersehen wir, dass der allgemeine Ausdruck einer Ableitungsfunktion im Zeileneditor oder Stack wie folgt lautet: ∫(*lower_limit, upper_limit,integrand,variable_of_integration*)
(unterer Grenzwert, oberer Grenzwert, Integrand, Integrationsvariable)

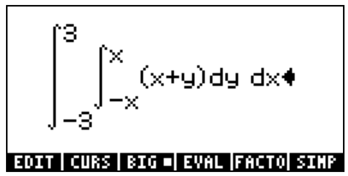
Drücken Sie [ENTER], um zum EquationWriter zurückzukehren. Die entstandene Anzeige ist nicht die bestimmte Integrale, die wir eingegeben haben, sondern deren symbolischer Wert, und zwar:



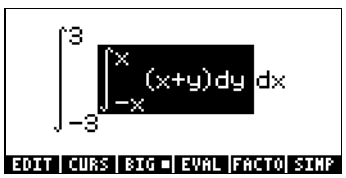
Um die Ableitungsfunktion wieder herzustellen, verwenden Sie [↱] UNDO . Um die Ableitungsfunktion neu zu berechnen, verwenden Sie die Funktionstaste [F4] . Diese zeigt erneut, dass

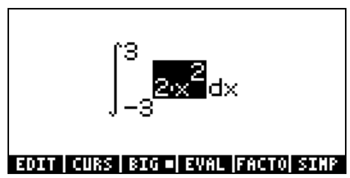
$$\int_0^{\tau} t \cdot \sin(t) \cdot dt = \sin(\tau) - \tau \cdot \cos(\tau)$$

Auch Doppelintegrale sind möglich. So zum Beispiel:



welche ausgewertet 36 ergibt. Teilauswertung ist auch möglich, so z.B.:

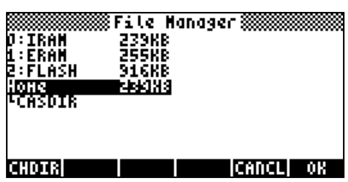




Der Wert dieser Integrale beträgt 36.

Organisieren der Daten im Rechner

Sie können Daten in Ihrem Rechner organisieren, indem Sie Variablen in einer Verzeichnisstruktur ablegen (speichern). Um den Speicher des Rechners zu verstehen, sehen wir uns erst einmal das Datenverzeichnis an. Drücken Sie die Tastenkombination ← FILES (erste Taste in der zweiten Reihe von oben), um in den Datenmanager des Rechners zu gelangen:



Diese Ansicht ist ein Schnappschuss des Rechnerspeichers und der Verzeichnisstruktur. In der Anzeige sehen wir, dass der Rechner drei

Speicherschnittstellen (oder Speicherpartitionen) hat, Schnittstelle *0:IRAM*, Schnittstelle *1:ERAM*, und Schnittstelle *2:FLASH* . Speicherschnittsellen werden dazu benutzt Anwendungen Drittanbieter oder Bibliotheken, aber auch um Backups zu speichern. Auch die Größe der drei unterschiedlichen Schnittstellen wird angegeben. In der vierten und den darauf folgenden Zeilen der Anzeige wird die Verzeichnisstruktur des Rechners angezeigt. Das oberste Verzeichnis (im Moment hervorgehoben) ist das *Home* Verzeichnis und enthält ein vordefiniertes Unterverzeichnis *CASDIR*. Die File Manager (Datei-Manager) Ansicht hat drei Funktionen, die Funktionstasten zugeordnet sind:

CHDIR (F1): Wechselt in das ausgewählte Verzeichnis
CANCL (F5): Aktion abbrechen
OK (F6): Auswahl bestätigen

Um z.B. ins Verzeichnish CASDIR zu wechseln, drücken Sie die Pfeiltaste ▼ und aschließend CHDIR (F1). Dadurch wird das *File Manager* Fenster geschlossen und wir erhalten die Normalanzeige des Rechners. In der zweiten Zeile von oben werden Sie in der Anzeige { HOME CASDIR } sehen, welches Ihnen anzeigt, dass Sie sich im CASDIR innerhalb des HOME Verzeichnisses befinden.

Funktionen zur Manipulation von Variablen

In dieser Anzeige stehen insgesamt 20 Befehle zur Verfügung, welche Funktionstasten zugeordnet sind und der Erstellung, Bearbeitung und Manipulation von Variablen dienen. Die ersten sechs Funktionen sind wie folgt:

EDIT zum bearbeiten einer hervorgehobenen Variablen
COPY zum kopieren einer hervorgehobenen Variablen
MOVE zum verschieben einer hervogehobenen Variablen
RCL zum wiederherstellen des Inhalts einer hervorgehobenen Variablen
EVAL zum berechnen einer hervogehobenen Variablen
TREE zum anzeigen der Verzeichnisstruktur in der sich die Variable befindet

Wenn Sie die Taste NXT drücken, erhalten Sie das nächste zur Verfügung stehende Funktionsset:

PURGE zum löschen oder bereinigen einer Variablen

RENAM zum umbenennen einer Variablen
NEW zum erstellen einer neuen Variablen
ORDER zum anordnen mehrerer Variablen in einem Verzeichnis
SEND zum senden einer Variablen an einen anderen Rechner oder Computer
RECV zum empfangen einer Variablen von einem anderen Rechner oder Computer

Wenn Sie die Taste NXT drücken, erhalten Sie das dritte zur Verfügung stehende Funktionsset:

HALT um vorrübergehend zum Stack zurückzukehren
VIEW um den Inhalt einer Variablen anzuzeigen
EDITB um den Inhalt einer binären Variablen zu bearbeiten (ähnlich wie EDIT)
HEADE um das Verzeichnis, in dem sich die Variable aus der Kopfzeile befindet, anzuzeigen
LIST stellt eine Liste von Variablennamen und Beschreibungen bereit
SORT um Variablen nach einem bestimmten Sortierkriterium zu anzuordnen

Wenn Sie die Taste NXT drücken, erhalten Sie das letzte zur Verfügung stehende Funktionsset:

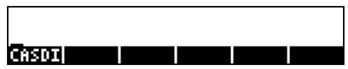
XSEND um eine Variable mit X-Modem Protokoll zu senden
CHDIR um das Verzeichnis zu wechseln

Um zwischen den verschiedenen Funktionsmenü-Befehlen zu navigieren, können Sie nicht nur die Taste NEXT (NXT), sondern auch die Taste PREV (← PREV) verwenden.

Der Benutzer soll nun diese Funktionen selbst ausprobieren. Deren Anwendung ist ziemlich einfach.

Das HOME Verzeichnis

Wie schon vorhin erwähnt, ist das HOME Verzeichnis, das Basis-Verzeichnis des Rechners. Um ins HOME Verzeichnis zu gelangen, können Sie die Funktion UPDIR (← UPDIR) benutzen – wiederholen Sie diesen Vorgang solange bis der Ausdruck { HOME } in der zweiten Zeile Ihres Displays erscheint. Alternativ dazu können Sie auch die ←(halten) UPDIR verwenden, drücken Sie ENTER im algebraischen Modus. In diesem Beispiel, enthält das HOME Verzeichnis nichts weiter als das CASDIR (CAS-Verzeichnis). Drücken Sie die Taste VAR, werden die Variablen der Funktionstasten angezeigt:

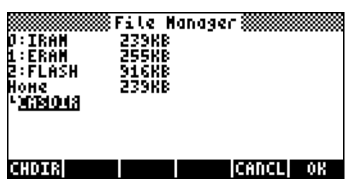


Unterverzeichnisse

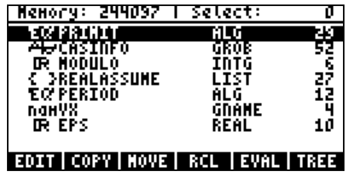
Um Ihre Daten in einer gut organisierten Verzeichnisstruktur zu speichern, können Sie Unterverzeichnisse im HOME Verzeichnis anlegen und weitere Unterverzeichnisse in diesen, in einer Hierarchie, ähnlich wie diese in Form von Ordnern auf modernen Computern zu finden ist. Die Unterverzeichnisse können so benannt werden, dass der Name bereits über den Inhalt des Unterverzeichnisses Auskunft gibt, oder aber kann es ein beliebiger Name, der Ihnen gerade einfällt, sein.

Das Unterverzeichnis CASDIR

Das Unterverzeichnis CASDIR enthält einige, für die korrekte Funktion des Rechners notwendigen Variablen des CAS (Computer Algebraisches System, siehe Anhang C). Um sich den Inhalt des Verzeichnisses anzeigen zu lassen, können wir die Tastenkombination: ← FILES benutzen, welche den *File Manager* (Dateimanager) erneut öffnet:



Diesmal ist CASDIR in der Anzeige hervorgehoben. Um den Inhalt des Verzeichnisses anzuzeigen, drücken Sie die Funktionstaste OK (F6) oder ENTER, um die folgende Ansicht zu erhalten:



In der Anzeige ist eine Tabelle, die die Variablen im CASDIR beschreibt. Dies sind im Speicher des Rechners vordefinierte Variablen, welche bestimmte Parameter für die CAS Operation festlegen (siehe Anhang C). Die obige Tabelle enthält vier Spalten:

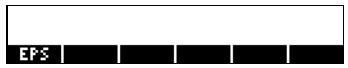
- In der ersten Spalte ist der Typ der Variablen angezeigt (z.B. 'EQ' bedeutet, eine Variable des Typs Gleichung, |R zeigt an, dass es sich um eine Variable mit einem reellem Wert handelt, { } bedeutet eine Liste, *nam* bedeutet 'ein globaler Name' und das Symbol stellt eine grafische Variable dar.
- Die zweite Spalte stellt den Namen der Variablen, d.h. *PRIMIT, CASINFO, MODULO, REALASSUME, PERIOD, VX* und *EPS* dar.
- Spalte 3 weist eine weitere Spezifikation für den Typ der Variablen, z.B. *ALG* bedeutet algebraischer Ausdruck, *GROB* steht für grafisches Objekt, *INTG* bedeutet eine numerische Ganzzahl-Variable, *LIST* stellt eine Liste von Daten dar, *GNAME* bedeutet globaler Name und *REAL* ist eine reelle (oder Gleitzahlkomma) numerische Variable.
- Die vierte und letzte Spalte ist die Größe in Bytes, der abgeschnittenen Variablen ohne Dezimalstellen (d.h. Halbbyte). So z.B. benötigt die Variable PERIOD 12,5 Bytes während die Variable REALASSUME 27,5 Bytes braucht (1 Byte = 8 Bits, 1 Bit ist die kleinste Einheit im Speicher von Computern und Rechnern).

CASDIR Variablen im Stack

Drücken Sie die Taste ON wird die vorangegangene Anzeige geschlossen und Sie erhalten die Normalanzeige des Rechners. Standardmäßig kommen wir zum Menü TOOL zurück:

Wir können die Variablen im aktuellen Verzeichnisn CASDIR ansehen, indem wir die Taste VAR drücken (erste Taste in der zweiten Reihe von oben). Folgende Anzeige erscheint:

Drücken Sie die Taste NXT, sehen Sie eine weitere, in diesem Verzeichnis gespeicherte Variable:



- Um den Inhalt der Variablen EPS z.B. anzuzeigen, drücken Sie ↱ EPS. Diese zeigt den Wert von EPS als .0000000001 an.
- Um den Wert einer numerischen Variablen anzuzeigen, müssen wir nur die entsprechende Funktionstaste für die Variable drücken. So z.B. drücken wir EPS gefolgt von ENTER, zeigt den gleichen Wert der Variablen im Stack, wenn der Rechner auf *Algebraic* gesetzt ist. Befindet sich der Rechner im *RPN*-Modus, müssen Sie nur die Funktionstaste für ENTER drücken.
- Um den vollen Namen einer Variablen anzuzeigen, drücken Sie erst den Apostroph ' und dann díe der Variablen entsprechende Funktionstaste. So z.B. für die im Stack aufgelistete Variable PERIO verwenden wir: ' PERIO, welche den String 'PERIOD' ausgibt. Diese Prozedur ist für beide Operations-Modi des Rechners, den *algebraischen* wie auch den *RPN*-Modus, gültig.

Variablen in CASDIR

Die in CASDIR enthaltenen Standardvariablen sind:

PRIMIT	Letzte berechnete Stammfunktion, keine Standardvariable, sondern eine die wir in einem vorangegangenen Beispiel erstellt haben
CASINFO	ein Graph das CAS Informationen liefert
MODULO	Modulo für modulare Arithmetik (Standard = 13)
REALASSUME	Auflistung von Variablennamen, von welchen angenommen wird, dass sie reelle Werte darstellen
PERIOD	Intervall für trigonometrische Funktionen (Stadard = 2π)
VX	Name der unabhängigen Standardvariablen (Standard = X)
EPS	Wert des kleinen Inkrementes (Epsilon), (Standard = 10^{-10})

Diese Variablen werden für die Funktion des CAS benutzt.

Verzeichnis- und Variablen-Namen tippen

Um Namen für Variablen einzugeben, müssen Sie eine Zeichenfolge auf einmal eingeben, welche entweder nur aus Buchstaben oder aus einer Kombination von Buchstaben und Zahlen bestehen kann. Wenn Sie nicht die Tasten [ALPHA], oder Tastenkombination [ALPHA][←] oder [ALPHA][→] beim eingeben jedes Buchstabens betätigen möchten, können Sie auch nur die Taste [ALPHA] halten und den Namen einfach eingeben. Sie können aber auch die alphabetische Tastatur temporär feststellen und einen kompletten Namen eingeben, bevor Sie diese wieder freigeben. Folgende Tastenkombinationen stellen die alphabetische Tastatur fest:

[ALPHA][ALPHA] stellt die alphabetische Tastatur in Großbuchstaben fest. Wenn die Tastatur so fest eingestellt ist, können Sie Kleinbuchstaben mit gehaltener [←] Taste erzeugen, während Sie Sonderzeichen mit gehaltener [→] Taste eingeben können. Ist die alphabetische Tastatur in Großbuchstaben festgestellt, können Sie diese in Kleinbuchstaben mit der Tastenkombination [←][ALPHA] umändern.

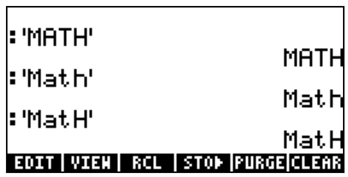
[ALPHA][ALPHA][←][ALPHA] stellt die alphabetische Tastatur auf Kleinbuchstaben fest. So eingestellt, müssen Sie, bevor Sie einen Großbuchstaben eingeben wollen, die Taste [←] drücken, Um die Eingabe für Kleinbuchstaben rückgängig zu machen, drücken Sie [←][ALPHA]

Um die Eingabe für Großbuchstaben rückgängig zu machen, drücken Sie [ALPHA]

Versuchen wir nun einige Beispiele Verzeichnisse/Variablen Namen in den Stack einzugeben. Angenommen Sie befinden sich im algebraischen Modus (obwohl diese Anweisungen genauso im RPN-Modus funktionieren), versuchen Sie die nachfolgende Tastenkombination. Mit nachfongenden Anweisungen werden wir die folgenden Wörter eingeben 'MATH', 'Math' und 'MatH'

['][ALPHA][ALPHA][M][A][T][H][/][ALPHA][ENTER]
['][ALPHA][ALPHA][M][←][A][←][T][←][H][ALPHA][ENTER]
['][ALPHA][ALPHA][M][←][ALPHA][A][T][←][H][ALPHA][ENTER]

Im Display des Rechners wird folgendes angezeigt (links algebraischer Modus, rechts RPN-Modus):



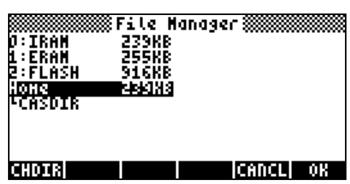
Anmerkung: Wenn System Flag 60 gesetzt ist, können Sie die alphabetische Tastatur einfach mit der Taste ALPHA feststellen. Weitere Informationen zu System Flags finden Sie in Kapitel 1.

Erstellen von Unterverzeichnissen

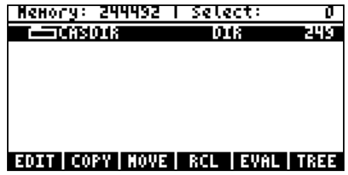
Unterverzeichnisse können entweder in der FILES Umgebung oder mit dem Befehl CRDIR erstellt werden. Nachfolgend werden beide Ansätze zur Erstellung von Unterverzeichnissen vorgestellt.

Verwenden des Menüs FILES

Unabhängig vom Operationsmodus des Rechners (algebraisch oder RPN), können wir eine Verzeichnisstruktur, basierend auf dem HOME Verzeichnis, anhand der im FILES Menü aktivierten Funktionen, erstellen. Drücken Sie ← FILES , um das Menü FILES zu starten. Sofern das HOME Verzeichnis noch nicht hervorgehoben ist, d.h.



benutzen Sie die Pfeiltasten (▲▼), um dieses hervorzuheben. Drücken Sie anschließend die Funktionstaste F6 (OK) . Die Anzeige sieht in etwa so aus:



wobei Sie feststellen können, dass sich zur Zeit nur ein Objekt im HOME Verzeichnis, und zwar das Unterverzeichnis CASDIR, befindet. Erstellen wir nun ein weiteres Unterverzeichnis mit den Namen MANS (für Handbücher – MANualS) in welchem wir die Variablen, welche wir im Laufe dieses Handbuchs erstellen, speichern wollen. Um dieses Unterverzeichnis zu erstellen, drücken Sie erst: NXT NEW (F3). Sie erhalten die folgende Eingabemaske:

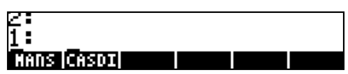
Das Eingabefeld *Object*, das erste Eingabefeld in der Maske und ist standardmäßig hervorgehoben. Dieses Feld kann den Inhalt einer neuen Variablen, die wir erstellen, enthalten. Da es noch keine Inhalte für das neue Unterverzeichnis an dieser Stelle gibt, überspringen wir dieses Eingabefeld einfach mit der Pfeiltaste ▼ einmal. Nun wird das Feld *Name* hervorgehoben.

An dieser Stelle geben wir den Namen des neuen Unterverzeichnisses (oder der Variablen, welches auch der Fall sein mag) wie folgt ein:
ALPHA ALPHA M A N S ENTER

Der Cursor springt ins Kontrollfeld *_Directory*. Drücken Sie die Funktionstaste CHK (F3) um anzugeben, dass Sie ein Verzeichnis erstellen und anschließend OK, um die Eingabemaske zu verlassen.   Die Variablenauflistung für das HOME Verzeichnis können Sie in der nachfolgenden Anzeige sehen:

In der Anzeige sehen Sie, dass ein neues Verzeichnis (MANS) innerhalb des HOME Verzeichnisses existiert.

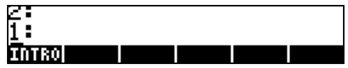
Als nächstes erstellen wir ein Unterverzeichnis mit dem Namen INTRO (für Einführung – INTROduction) innerhalb des Verzeichnisses MANS, um die Variablen der Beispiele der nächsten Abschnitte dieses Kapitels zu speichern. Drücken Sie die Taste ON, um zur Normalanzeige des Rechners (das Menü TOOLS wird angezeigt) zurückzukehren. Drücken Sie dann VAR, um die Inhalte des HOME Verzeichnisses in den Funktionstasten anzuzeigen Die Anzeige könnte wie folgt aussehen (haben Sie weitere Variablen innerhalb des HOME Verzeichnisses erstellt, werden diese ebenfalls bei den Funktionstasten angezeigt):



Um in MANS Verzeichnis zu wechseln, drücken Sie die entsprechende Funktionstaste (in diesem Fall die F1) und, falls im algebraischen Modus, die Taste ENTER. In der zweiten Zeile der Verzeichnisstruktur wird { HOME MANS } angezeigt. Die Funktionstasten aber, werden keine Beschriftung, wie unten gezeigt aufweisen, weil noch keine Variablen für dieses Verzeichnis gespeichert wurden.
Erstellen wir nun ein neues Unterverzeichnis INTRO, indem wir wie folgt vorgehen:

↰ FILES OK NXT NEW ▼ ALPHA ALPHA I N T R O ENTER CHK OK

Drücken Sie die Taste ON gefolgt von der Taste VAR, um sich den Inhalt des Verzeichnisses MANS wie folgt anzuzeigen:



Drücken Sie nun die Funktionstaste INTRO, um ins Unterverzeichnis INTRO zu wechseln. Ein leeres Verzeichnis wird angezeigt. Später werden wir einige Beispiele zur Erstellung von Variablen erzeugen.

Verwenden des Befehls CRDIR

Mit dem Befehl CRDIR können Verzeichnisse erstellt werden. Diesen Befehl erreicht man über die Tasten Befehle Katalog (die Taste →_CAT , zweite in der vierten Reihe von oben), über die Programmiermenüs (die Taste ←_PRG , entspricht der gleichen Funktion wie die Taste →_CAT) oder einfach durch eintippen dieser.

- Über die Taste Katalog
  Drücken Sie →_CAT ALPHA C . Benutzen Sie die Pfeiltasten (▲▼), um CRDIR zu finden. Drücken Sie die Funktionstaste OK, um den Befehl zu starten.
- Über die Programmiermenüs
  Drücken Sie ←_PRG . Dadurch erhalten Sie das nachfolgende Aktionsmenü zur Programmierung:

Benutzen Sie die Pfeiltaste (▼), um Option *2. MEMORY…* auszuwählen oder einfach nur die 2. Drücken Sie anschließend OK. Dadurch erhalten Sie das nachfolgende Pull-Down Menü:

Benutzen Sie die Pfeiltaste (▼), um Option *5. MEMORY…* auszuwählen oder einfach nur die 5 . Drücken Sie anschließend OK. Dadurch erhalten Sie das nachfolgende Pull-Down Menü:

Benutzen Sie die Pfeiltaste (▼), um Option *5. DRDIR…* auszuwählen und dann OK.

Befehl DRDIR im algebraischen Modus

Sobald Sie mit einer der obengenannten Möglichkeiten das CRDIR ausgewählt haben, steht Ihnen dieser Befehl im Stack wie folgt zur Verfügung:

An dieser Stelle, müssen Sie einen Verzeichnisnamen, sagen wir *chap1*, eingeben:

ALPHA ALPHA ← ALPHA C H A P 1 ALPHA ENTER

Der Name des neuen Verzeichnisses wird im Funktionstastenmenü angezeigt, z.B.

Befehl DRDIR im RPN-Modus

Um CRDIR im RPN-Modus zu benutzen, muss, bevor Sie den Befehl starten, der Name des Verzeichnisses im Stack bereits existieren. So zum Beispiel:

ALPHA ALPHA ← ALPHA C H A P 2 ALPHA ENTER

Starten Sie den Befehl CRDIR mit einer der oben genannten Möglichkeiten, z.B. über die Taste → CAT :

Drücken Sie die Funktionstaste OK, um den Befehl zur Erstellung des Unterverzeichnisses zu aktivieren:

Zwischen den Unterveichnissen hin und her wechseln

Um in der Verzeichnisstruktur weiter nach unten zu wechseln, müssen Sie die, dem Unterverzeichnis entsprechende Funktionstaste, in welches Sie wechseln möchten, eintippen. Die Auflistung der Variablen in einem Unterverzeichnis, erhalten Sie, wenn Sie die Taste VAR (VARiablen) drücken. Um in ein übergeordnetes Verzeichnis der Funktion zu wechseln, benutzen Sie die Funktion UPDIR, d.h. Sie geben ← UPDIR ein.

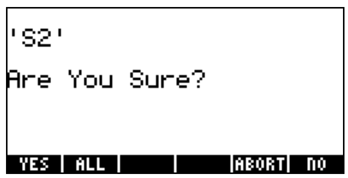
Alternativ, können Sie auch das Menü FILES dazu benutzen, d.h. Sie drücken ← FILES . Benutzen Sie die Pfeiltasten (▲▼), um das Unterverzeichnis, in welches Sie wechseln wollen auszuwählen, und anschließend CHDIR (Change DIRectory) oder die Funktionstaste F1 . Dadurch werden Ihnen die Inhalte des Unterverzeichnisses, in welches Sie gewechselt haben, im Funktionstastenmenü angezeigt.

Löschen von Unterverzeichnissen

Um ein Unterverzeichnis zu löschen, verwenden Sie eine der nachfolgenden Prozeduren:

Verwenden des Menüs FILES

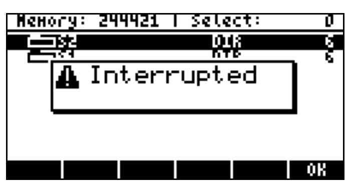
Drücken Sie die Taste ← FILES , um das FILES Menü auszuwählen. Wählen Sie das Verzeichnis, in welchem sich das Unterverzeichnis, das Sie löschen möchten, befindet und drücken dann, falls nötig CHDIR . Das FILES Menü wird geschlossen und die Inhalte des ausgewählten Verzeichnisses angezeigt. In diesem Fall müssen Sie die Taste ENTER drücken. Drücken Sie die Funktionstaste OK , um sich die Inhalte des Verzeichnisses auf dem Display anzuzeigen. Wählen Sie das Unterverzeichnis (oder die Variable), das Sie löschen möchten, aus. Drücken Sie NXT PURGE. Sie sehen eine ähnliche Anzeige, wie unten:



Die Zeichenfolge 'S2' ist in diesem Fall der Name des zu löschenden Unterverzeichnisses. Das Funktionstastenmenü enthält folgende Optionen:

YES (F1) Fahren Sie mit dem löschen des Unterverzeichnisses (der Variablen) fort
ALL (F2) Fahren Sie mit dem löschen aller Unterverzeichnisse (Variablen) fort
ABORT (F5) Unterverzeichnis (Variable) nicht aus einer Liste löschen
NO (F6) Unterverzeichnis (Variable) nicht löschen

Nachdem Sie nun einen dieser vier Befehle ausgewählt haben, kommen Sie zur Anzeige der Inhalte des Unterverzeichnisses zurück. Der Befehl ABORT, bringt jedoch eine Fehlermeldung:



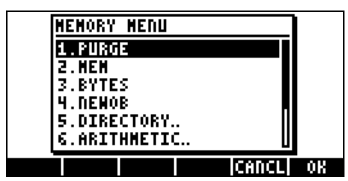
und Sie müssen die Taste OK drücken, bevor Sie zur Auflistung der Variablen zurückkehren.

Verwenden des Befehls PGDIR

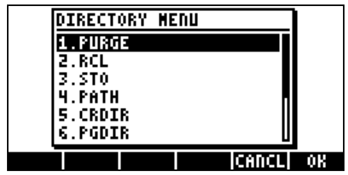
Mit dem Befehl PGDIR können Verzeichnisse bereinigt werden. Genau wie der Befehl CRDIR, ist der Befehl PGDIR über die Taste ‹CAT› oder über die Taste ‹PRG› auswählbar oder kann direkt eingegeben werden.

- Über die Taste Katalog
  Drücken Sie ‹CAT› ‹ALPHA›‹ALPHA›‹P›‹G›. Der Befehl PGDIR sollte nun hervorgehoben sein. Drücken Sie die Funktionstaste OK, um den Befehl zu starten.
- Über die Programmiermenüs
  Drücken Sie ‹PRG›. Dadurch erhalten Sie das nachfolgende Aktionsmenü zur Programmierung:

Benutzen Sie die Pfeiltaste (▼), um Option *2. MEMORY…* auszuwählen und dann OK. Dadurch erhalten Sie das nachfolgende Pull-Down Menü:



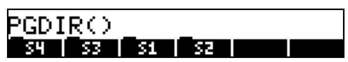
Benutzen Sie die Pfeiltaste (▼), um Option *5. DIRECTORY …* auszuwählen. Drücken Sie anschließend OK. Dadurch erhalten Sie das nachfolgende Pull-Down Menü:



Benutzen Sie die Pfeiltaste (▽), um Option *6. PGDIR…* auszuwählen und dann OK.

Befehl PGDIR im algebraischen Modus

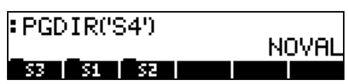
Sobald Sie mit einer der obengenannten Möglichkeiten das PGDIR ausgewählt haben, steht Ihnen dieser Befehl im Stack wie folgt zur Verfügung:



An dieser Stelle müssen Sie den Namen eines existierenden Verzeichnisses eingeben, sagen wir *S4*:

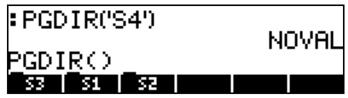
ALPHA S 4 ENTER

Als Ergebnis wird das Unterverzeichnis S4 gelöscht:

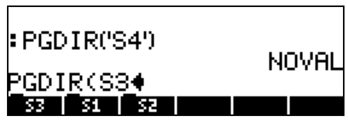


Anstatt den Namen des Verzeichnisses einzutippen, können Sie auch einfach die entsprechende Funktionstaste aus der Auflistung der PGDIR ( ) Befehle drücken, z.B.

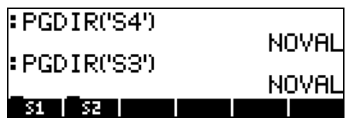
Drücken Sie OK, um nachfolgendes zu erhalten:



Anschließend drücken Sie S3, um 'S3' als das Argument zu PGDIR einzugeben.



Drücken Sie ENTER, um das Unterverzeichnis zu löschen:

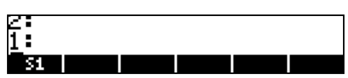


Befehl PGDIR im RPN-Modus

Um PGDIR im RPN-Modus zu benutzen, muss, bevor Sie den Befehl starten, der Name des Verzeichnisses im Stack bereits existieren. So zum Beispiel:
' ALPHA S 2 ENTER

Starten Sie den Befehl PGDIR mit einer der oben genannten Möglichkeiten, z.B. über die Taste ↱ CAT :

Drücken Sie die Funktionstaste OK, um den Befehl zur Löschung des Unterverzeichnisses zu aktivieren:



Anwendung des PURGE Befehls aus dem Menü TOOL

Das Menü TOOL wird durch drücken der Taste TOOL (algebraischer und RPN-Modus werden angezeigt):

Der Befehl PURGE kann über die Funktionstaste PURGE (F5) aktiviert werden. In den nachfolgenden Beispielen wollen wir das Unterverzeichnis *S1* löschen:

- Algebraischer Modus: Drücken Sie PURGE VAR S1 ENTER
- RPN-Modus: Drücken Sie VAR ' S1 ENTER TOOL PURGE VAR

Variablen

Variablen sind ähnlich wie Dateien auf der Festplatte eines Computers. Eine Variable kann ein Objekt speichern (numerische Werte, algebraische Ausdrücke, Listen, Vektoren, Matrizen, Programme usw.). Auch Unterverzeichnisse können als Variablen dargestellt werden (eigentlich ist ein Unterverzeichnis im Rechner gleichzeitig eine Art Rechner-Objekt).

Variablen werden über deren Namen aufgerufen, welche aus einer beliebigen Kombination von Buchstaben und Zahlen sein kann, wobei aber der Anfangsbuchstabe immer ein Buchstabe sein muss (englisch oder griechisch). Einige Sonderzeichen, wie z.B. der Pfeil (→) kann im Variablennamen verwendet werden, aber nur in Kombination mit einem Buchstaben. Somit ist '→A' ein gültiger Name für eine Variable, '→' hingegen nicht. Beispiele von gültigen Variablennamen sind: 'A', 'B', 'a', 'b', 'α', 'β', 'A1', 'AB12', '→A12','Vel','Z0','z1', usw.

Eine Variable kann nicht denselben Namen wie eine Funktion im Rechner haben. Sie können also keine Variable mit den Namen SIN erstellen, weil ein Befehl mit dem Namen SIN im Rechner existiert. Nachfolgend ist eine Auflistung reservierter Variablennamen im Rechner: ALRMDAT, CST, EQ, EXPR, IERR, IOPAR, MAXR, MINR, PICT, PPAR, PRTPAR, VPAR, ZPAR, der_, e, i, n1,n2, …, s1, s2, …, ΣDAT, ΣPAR, π, ∞

Variablen können in Unterverzeichnissen organisiert werden.

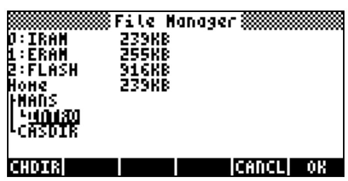
Erstellen von Variablen

Genau wie wir dies mit Unterverzeichnissen gemacht haben, lassen sich auch Variablen über das Menü FILES erstellen. So z.B. wollen wir in einem vorher erstellten Unterverzeichnis {HOME MANS INTRO} nachfolgende Variablen mit den unten aufgeführten Werten erstellen .

Name	***Inhalt***	***Typ***
A	12.5	reell
α	-0.25	reell
A12	3×10^5	reell
Q	'r/(m+r)'	Algebraik
R	[3,2,1]	Vektor
z1	3+5i	komplex
p1	<< → r 'π*r^2' >>	Programm

Verwenden des Menüs FILES

Benutzen wir das Menü FILES, um die Variable A einzugeben. Nehmen wir an, dass wir uns im Unterverzeichnis {HOME MANS INTRO} befinden. Um in dieses Unterverzeichnis zu gelangen, drücken Sie: ← FILES und wählen Sie das Unterverzeichnis INTERO, wie unten gezeigt:



Drücken Sie OK um ins Verzeichnis zu gelangen. Sie bekommen eine Anzeige "no entries" – keine Einträge (das Unterverzeichnis INTRO ist an dieser Stelle noch leer)

Drücken Sie die Taste NXT, um zum nächsten Funktionstastenmenü zu gelangen und drücken Sie die Funktionstaste NEW. Sie erhalten die folgende Eingabemaske für NEW VARIABLE (neue Variable):

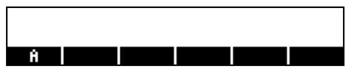
Um die Variable A (siehe Tabelle unten) einzugeben, müssen wir zuerst den Inhalt dieser eingeben, und zwar die Zahl 12,5 und anschließend deren Namen, wie folgt: 1 2 . 5
OK ALPHA A OK. Die Anzeige sieht wie folgt aus:

Drücken Sie OK ein weiteres Mal, um die Variable zu erstellen. Die neue Variable wird wie folgt in der Auflistung angezeigt:

Die Auflistung zeigt eine reelle Variable (|R) mit dem Namen A, welche einen Speicherplatz von 10,5 Byte belegt. Um sich den Inhalt der Variablen anzuzeigen, drücken Sie NXT VIEW.

- Drücken Sie die Funktionstaste GRAPH (F1) um den Inhalt in grafischer Form darzustellen.
-

- Drücken Sie die Funktionstaste TEXT (F1) um den Inhalt in Textform darzustellen.
- Drücken Sie OK, um zur Variablenliste zurückzukehren
- Drücken Sie ON ein weiteres Mal, um zur Normalanzeige zurückzukehren. Die Variable A sollte nun bei den Funktionstasten angezeigt werden:



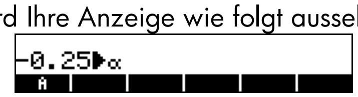
Verwenden des Befehls STO▶

Ein einfacherer Weg eine Variable zu erstellen, ist über den Befehl STO (d.h. die Taste STO▶). Für die Erstellung der noch fehlenden Variablen, werden die Beispiele im algebraischen und im RPN-Modus angezeigt.

Name	***Inhalt***	***Typ***
α	-0.25	reell
A12	3×10^5	reell
Q	‘r/(m+r)'	Algebraik
R	[3,2,1]	Vektor
z1	3+5i	komplex
p1	<< → r 'π*r^2' >>	Programm

Algebraischer Modus

Drücken Sie nachfolgende Tastenfolge, um den Wert -0,25 in eine Variable α zu speichern: 0 . 2 5 +/- STO▶ ALPHA ↱ A .
An dieser Stelle wird Ihre Anzeige wie folgt aussehen:



Dieser Ausdruck bedeutet, dass der Wert -0,25 in α gespeichert wird (das Symbol ▶ deutet darauf hin). Drücken Sie nun ENTER, um die

Variable zu erzeugen. Die Variable wird nun bei den Funktionstasten angezeigt.

Nachfolgend die Tastenkombinationen zur Eingabe der noch verbleibenden Variablen:

A12: 3 EEX 5 STO▶ ALPHA A 1 2 ENTER

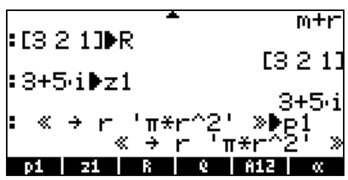
Q: ' ALPHA ← R ÷ ← () ALPHA ← M + ALPHA ← R ▶ ▶ STO▶ ALPHA Q ENTER

R: ← [] 3 → , 2 → , 1 ▶ STO▶ ALPHA R ENTER

z1: 3 + 5 × ← i STO▶ ALPHA ← Z 1 ENTER (Bestätigen Sie den Wechsel in den *Complex Modus, falls gefragt*).

p1: → « » → → ALPHA ← R ' ← π × ALPHA ← R Y^x 2 ▶ ▶ ▶ STO▶ ALPHA ← P 1 ENTER ..

An dieser Stelle, sieht Ihre Anzeige wie folgt aus:

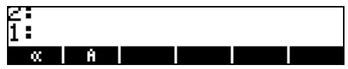


Am unteren Rand werden Sie sechs oder sieben Variablen feststellen: *p1, z1, R, Q, A12, α*.

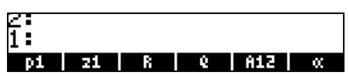
RPN-Modus

Drücken Sie nachfolgende Tastenfolge, um den Wert -0,25 in eine Variable α zu speichern: 0 . 2 5 +/- ENTER ALPHA → A ENTER . An dieser Stelle wird Ihre Anzeige wie folgt aussehen:

Dieser Ausdruck bedeutet, dass der Wert -0,25 bereit zur Speicherung nach α ist. Drücken Sie nun STO▶, um die Variable zu erzeugen. Die Variable wird nun bei den Funktionstasten angezeigt.



Um den Wert 3×10^5 in die Variable A12 einzugeben, kann auch eine kürzere Version der Prozedur benutzt werden:
3 EEX 5 ' ALPHA A 1 2 ENTER STO▶
Nachfolgend eine Methode den Inhalt von Q einzugeben:
Q: ' ALPHA ← R ÷ ← ()
ALPHA ← M + ALPHA ← R ▶ ▶ ' ALPHA Q ENTER STO▶
Um den Wert von R einzugeben, kann auch eine kürzere Prozedur benutzt werden:
R: ← [] 3 SPC 2 SPC 1 ▶ ' ALPHA R ENTER STO▶
Beachten Sie bitte, dass man die Elemente eines Vektors im RPN-Modus besser durch einen Leerschritt (SPC), als durch das Komma (→ ,) voneinander trennt, wie dies im algebraischen Modus verwendet wurde.
z1: ' 3 + 5 × ← i ' ALPHA ← Z 1 STO▶
(Bestätigen Sie den Wechsel in den *Complex Modus, falls gefragt*).
p1: → « » → → ALPHA ← R ' ← π ×
ALPHA ← R y^x 2 ▶ ▶ ▶ ' ALPHA ← P 1 ▶ ENTER STO▶.
An dieser Stelle, sieht Ihre Anzeige wie folgt aus:



Am unteren Rand werden Sie sechs oder sieben Variablen feststellen: *p1, z1, R, Q, A12,* α.

Überprüfen der Inhalte von Variablen

Als Übung in den Inhalt einer Variablen reinzuspähen, werden wir die sieben vorhin eingegeben Variablen benutzen. Wir haben gezeigt wie das FILES Menü dazu verwendet wird den Inhalt der Variablen A aus einer früheren

Übung anzusehen. In diesem Abschnitt zeigen wir einen einfacheren Weg in den Inhalt der Variablen reinzuschauen.

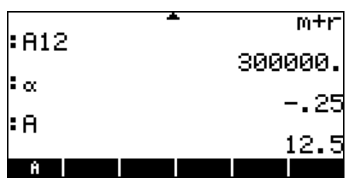
Funktionstaste für die Variable drücken

Diese Prozedur zeigt den Inhalt einer Variablen an, sofern diese einen numerischen, algebraischen Wert oder ein Array enthält.   So zum Beispiel können Sie zur Überprüfung des Inhalts der oben aufgeführten Variablen nachfolgende Tasten drücken:

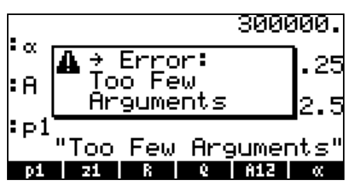
Algebraischer Modus

Drücken Sie nachfolgende Tastenfolge: VAR z1 ENTER R ENTER Q ENTER. An dieser Stelle sieht Ihre Anzeige wie folgt aus:

Als nächstes, drücken Sie nachfolgende Tastenfolge: A12 ENTER α ENTER NXT A ENTER. An dieser Stelle sieht Ihre Anzeige wie folgt aus:



Wenn Sie nun die Funktionstaste, die *p1* entsptricht, drücken erhalten Sie eine Fehlermeldung (versuchen Sie es mit NXT p1 ENTER)

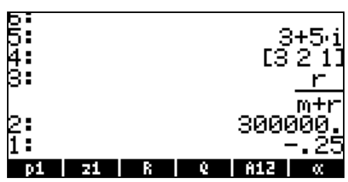


Anmerkung: Durch drücken von [p1] [ENTER] versuchen wir das Programm *p1* zu starten (run). Dieses Programm aber erwartet eine numerische Eingabe. Versuchen Sie folgendes Beispiel: [ON] [p1] [←][()] [5] [ENTER]. Die Lösung lautet:

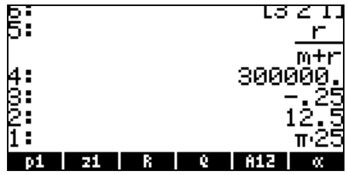
Das Programm hat folgende Struktur: << → r 'π*r^2' >>
Die Symbole « » weisen auf ein Programm in der RPL-Sprache hin (die originale Programmiersprache der HP 28/48 Rechner und auch in der 49G Reihe der HP Rechner vorhanden). Die Zeichen → r weisen darauf hin, dass eine Eingabe, welche als *r* bezeichnet wird, dem Programm zur Verfügung gestellt wird. Das Programm nimmt den Wert *r* und berechnet den algraischen Ausdruck 'π*r^2'. Im obigen Beispiel nimmt *r* den Wert 5 an, somit wird der Wert von πr^2 als π·25 wiedergegeben. Dieses Programm berechnet somit einer Kreisfläche mit einem gegebenen Radius von *r*.

RPN-Modus

Im RPN-Modus, müssen Sie nur die entsprechende Funktionstaste drücken um den Inhalt einer numerischen oder algebraischen Variablen zu erhalten. Im vorliegenden Fall, können wir versuchen in die oben erstellten Variablen *z1*, *R*, *Q*, *A12*, α, und *A*, wie folgt hineinzuspähen: [VAR] [z1] [R] [Q] [A12] [α]
An dieser Stelle sieht Ihre Anzeige wie folgt aus:



Um den Inhalt von A anzuzeigen, drücken Sie [NXT] [A]
Um das Programm *p1* mit r = 5 zu starten, drücken wir: [NXT] [5] [p1].



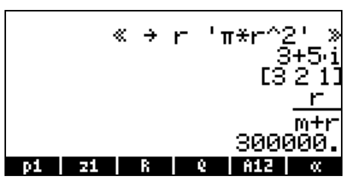
Beachten Sie, um das Programm im RPN-Modus auszuführen, müssen wir nun die Eingabe (5) machen und die entsprechende Funktionstaste drücken. (Im algebraischen Modus, müssen Klammern zur Eingabe des Argumentes gesetzt werden).

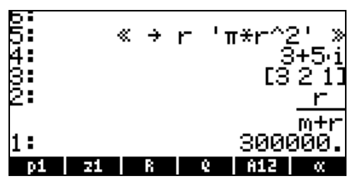
Verwendung der rechten Shift-Taste ↱ gefolgt von der entsprechenden Funktionstaste.

Dieser Ansatz den Inhalt einer Variablen anzusehen, funktioniert gleichermaßen im algebraischen wie auch im RPN-Modus. Versuchen Sie nachfolgende Beispiele in beiden Modi:

VAR ↱ p1 ↱ z1 ↱ R ↱ Q ↱ A12

Nach obiger Eingabe erscheint folgende Anzeige (algebraischer Modus links und RPN-Modus rechts)





Beachten Sie, dass in diesem Fall der Inhalt des Programms p1 in der Anzeige erscheint. Um die noch verbleibenden Variablen in diesem Verzeichnis anzusehen, gehen Sie folgendermaßen vor:

↱ α NXT ↱ i

Anzeigen der Inhalte aller Variablen im Bildschirm

Benutzen Sie die Tastenkombination ↱ ▼ um den Inhalt aller Variablen auf dem Display anzuzeigen. So zum Beispiel:

Drücken Sie die Taste ON, um zur Normalanzeige des Rechners zurückzukehren.

Inhalte von Variablen ersetzen

Der Austausch des Variableninhalts, ist als speichern eines unterschiedlichen Wertes in demselben Variablennamen zu betrachten. Somit, können wir die oben erstellten Variablen als Beispiele zur Veranschaulichung des Austauschs des Variableninhalts nehmen.

Verwenden des Befehls STO▶

Zur Veranschaulichung nehmen wir die sechs vorhin erstellten Variablen *p1, z1, R, Q, A12, a* und *A* und fahren fort den Inhalt der Variablen *A12* (im Moment eine numerische Variable) mit dem algebraischen Ausdruck 'β/2', unter Verwendung des Befehls STO▶ zu ändern. Zuerst im algebraischen Modus:

' ALPHA → β ÷ 2 ▶ STO▶ A12 ENTER

Überprüfen Sie den neuen Inhalt der Variablen *A12* mit Hilfe von → A12 .

Im RPN-Modus:

' ALPHA → β ÷ 2 ENTER ' A12 ENTER STO▶

oder vereinfacht

' ALPHA → β ÷ 2 ▶ ' A12 STO▶

Mit Hilfe der linken Shift-Taste ← gefolgt von der Funktionstaste der Variablen (RPN)

Dies ist eine äußerst einfache Art den Inhalt von Variablen zu ändern, funktioniert aber nur im RPN-Modus. Die Prozedur besteht darin, den neuen Inhalt der Variablen zu tippen und in den Stack einzugeben, anschließend die linke Shift-Taste, gefolgt von der der Variablen zugeordneten Funktionstaste zu drücken. So z.B., wenn wir im RPN-Modus den Inhalt der Variablen *z1* auf '*a+b·i* ' ändern möchten, verwenden wir:

' ALPHA ← A + ALPHA ← B × ← i ENTER

Dadurch wird der algebraische Ausdruck '*a+b·i* ' in Stack Ebene 1 eingetragen. Um dieses Ergebnis in die Variable *z1* einzugeben, verwenden wir: VAR ← z1

Um uns den Inhalt von *z1* anzusehen, verwenden wir: → z1

Ein äquivalenter Weg dies im algebraischen Modus zu tun ist folgender:

ALPHA ← A + ALPHA ← B × ← i ENTER STO▶ z1 ENTER

Um uns den neuen Inhalt von *z1* anzusehen, verwenden wir: → z1

Verwenden der Variablen ANS(1) (Algebraischer Modus)

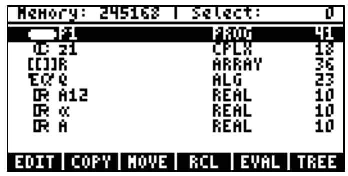
Im algebraischen Modus können wir die Variable ANS(1) verwenden,um den Inhalt einer Variablen auszutauschen. So z.B. die Prozedur, den Inhalt von *z1* auf '*a+bi'* abzuändern, ist folgende: ← ANS STO▶ z1 ENTER. Um uns den neuen Inhalt von *z1* anzusehen, verwenden wir: → z1

Kopieren von Variablen

Die nachfolgenden Übungen zeigen uns verschiedene Wege Variablen aus einem Unterverzeichnis in ein anderes zu kopieren.

Verwenden des Menüs FILES

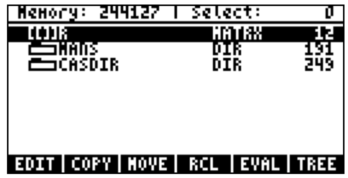
Um eine Variable von einem Unterverzeichnis in ein anderes zu kopieren können wir das Menü FILES verwenden. So haben wir z.B. im Unterverzeichnis {HOME MANS INTRO} folgende Variablen *p1, z1, R, Q, A12,* α und *A*. Angenommen wir wollen die Variable *A* kopieren und eine Kopie dieser im Unterverzeichnis {HOME MANS} ablegen. Weiterhin werden wir die Variable *R* kopieren und eine Kopie im HOME Verzeichnis ablegen. Nachfolgend wie es gemacht wird: Drücken Sie ← FILES OK, um nachfolgende Variablenliste zu erzeugen:



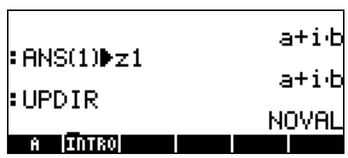
Benutzen Sie die Pfeiltaste ▼ um die Variable A auszuwählen (es ist die letzte in der Auflistung), anschließend drücken Sie COPY. Der Rechner wird sich mit der Anzeige 'PICK DESTINATION:' (Ziel wählen) melden

Benutzen Sie die Pfeiltaste ▲, um das Unterverzeichnis MANS auszuwählen und drücken Sie dann OK. Drücken Sie nun ← UPDIR, erscheint in der Anzeige der Inhalt des Unterverzeichnisses MANS (beachten Sie dass die Variable *A*, wie erwartet, in der Liste auftaucht):

Drücken Sie ON INTRO ENTER (im algebraischen Modus) oder ON INTRO (im RPN-Modus) um zum Verzeichnis INTRO zurückzukehren. Drücken Sie ← FILES OK, um die Variablenliste des Verzeichnisses {HOME MANS INTRO} zu erzeugen: Benutzen Sie die Pfeiltaste ▼, um die Variable *R* auszuwählen und dann COPY. Benutzen Sie die Pfeiltaste ▲, um das Verzeichnis HOME auszuwählen und drücken dann OK. Drücken Sie nun ← UPDIR zweimal, erscheint in der Anzeige das HOME Verzeichnis, in welchem sich eine Kopie von *R* befindet.



History im algebraischen Modus verwenden
Nachfolgend einen Weg um mit Hilfe von History (Stack) eine Variable aus einem Verzeichnis in ein anderes zu kopieren, wenn sich der Rechner im algebraischen Modus befindet. Angenommen wir befinden uns im Unterverzeichnis {HOME MANS INTRO} und möchten den Inhalt der Variablen *z1* ins Unterverzeichnis {HOME MANS} kopieren. Verwenden Sie dazu folgende Tastenfolge: → z1 STO▶ z1 ENTER Dadurch werden einfach die Inhalte von *z1* in sich selbst gespeichert (*z1* bleibt unverändert). Als nächstes, verwenden Sie ← UPDIR ENTER um ins {HOME MANS} Unterverzeichnis zu wechseln. Die Anzeige des Rechners sieht wie folgt aus:



Benutzen Sie die Löschaste ◀ ◀ ◀ (dreimal), um die letzten drei Zeilen im Display zu entfernen. An dieser Stelle ist der Stack bereit den Befehl ANS(1)▶z1 auszuführen. Drücken Sie ENTER, um den Befehl auszuführen. Verwenden Sie anschließend → z1, um den Inhalt der Variablen zu überprüfen.

Verwendung des Stacks im RPN-Modus
Um die Verwendung des Stacks beim kopieren einer Variable aus einem Unterverzeichnis in ein anderes zu demonstrieren, nehmen wir an Sie befinden sich im Unterverzeichnis {HOME MANS INTRO} und wir den Inhalt der Variablen *z1* ins HOME Verzeichnis kopieren wollen. Verwenden Sie nachfolgende Prozedur: → z1 ENTER ' z1 ENTER
Damit zeigen Sie den Inhalt und den Namen der Variablen im Stack an. Die Anzeige des Rechners sieht wie folgt aus:

Verwenden Sie nun [←] UPDIR [←] UPDIR um ins HOME Verzeichnis zu wechseln und drücken Sie [STO▶] um den Vorgang zu beenden. Mit [→] z1 können Sie den Inhalt der Variablen überprüfen.

Zwei oder mehrere Variablen im algebraischen Modus in den Stack kopieren
Nachfolgende Übung ist zur Demonstration des Kopiervorgangs zweier oder mehrerer Variablen über den Stack im algebraischen Modus gedacht. Angenommen wir befinden uns im Unterverzeichnis {HOME MANS INTRO} und möchten die Variablen *R* und *Q* ins Unterverzeichnis {HOME MANS} kopieren. Die dafür notwendigen Tastenaschläge, um diesen Vorgang abzuschließen sind wie folgt:

[→] R [STO▶] R [ENTER]
[→] Q [STO▶] Q [ENTER]
[←] UPDIR [ENTER]
[◀] [◀] [◀] [ENTER]
[◀] [◀] [◀] [◀] [ENTER]

Um den Inhalt der Variablen zu überprüfen, verwenden Sie [→] R and [→] Q.
Dieser Vorgang kann verallgemeinert werden, um drei oder mehrere Variablen zu kopieren.

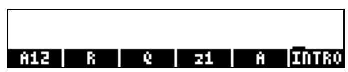
Zwei oder mehrere Variablen im RPN-Modus in den Stack kopieren
Nachfolgende Übung ist zur Demonstration des Kopiervorgangs zweier oder mehrerer Variablen über den Stack im RPN-Modus gedacht. Erneut nehmen wir an, dass wir uns im Unterverzeichnis {HOME MANS INTRO} befinden und die Variablen *R* und *Q* ins Unterverzeichnis {HOME MANS} kopieren möchten. Die dafür notwendigen Tastenaschläge, um diesen Vorgang abzuschließen sind wie folgt:

[→] R ['] R [ENTER]
[→] Q ['] Q [ENTER]
[←] UPDIR [STO▶] [STO▶]

Um den Inhalt der Variablen zu überprüfen, verwenden Sie ⟨→⟩ R and ⟨→⟩ Q.
Dieser Vorgang kann verallgemeinert werden, um drei oder mehrere Variablen zu kopieren.

Die Variablen in einem Verzeichnis neu anordnen

In diesem Abschnitt veranschaulichen wir, wie wir den Befehl ORDER zur Neuanordnung der Variablen in einem Verzeichnis verwenden. Nehmen wir an, dass wir vom Unterverzeichnis {HOMES MANS}, in welchem sich die Variablen *A12, R, Q, z1, A* und das Unterverzeichnis *INTRO* befinden, ausgehen - wie unten gezeigt. (Kopieren Sie A12 von INTRO zu MANS).

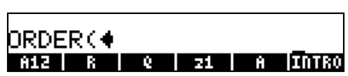


Algebraischer Modus

In diesem Fall ist der Rechner in den algebraischen Modus eingestellt. Angenommen wir wollen die Anordnung der Variablen auf *INTRO, A, z1, Q, R, A12* ändern. Um die Funktion ORDER zu starten, gehen Sie wie folgt vor:

⟨←⟩ PRG ⟨▼⟩ OK	Wählen Sie MEMORY aus dem Programmiermenü
⟨▼⟩⟨▼⟩⟨▼⟩⟨▼⟩ OK	Wählen Sie DIRECTORY aus dem Menü MEMORY
⟨▲⟩⟨▲⟩ OK	Wählen Sie ORDER aus dem Menü DIRECTORY

In der Anzeige wird die folgende Eingabezeile aufgezeigt:

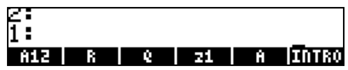


Als nächstes lassen wir uns die neue Anordnung der Variablen anzeigen, indem wir deren Namen in Anführungszeichen setzen:

⟨←⟩ { } ⟨'⟩ INTRO ⟨▶⟩ ⟨→⟩ , ⟨'⟩ A
⟨▶⟩ ⟨→⟩ , ⟨'⟩ z1 ⟨▶⟩ ⟨→⟩ , ⟨'⟩ Q ⟨▶⟩
⟨→⟩ , ⟨'⟩ R ⟨▶⟩ ⟨→⟩ , ⟨'⟩ A12 ENTER

Im Display sehen Sie die neue Anordnung der Variablen:

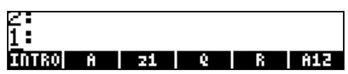
RPN-Modus
Im RPN-Modus, wird die Auflistung der neu angeordneten Variablen vor Anwendung des Befehls ORDER angezeigt. Angenommen wir beginnen mit derselben Situation wie oben, aber im RPN-Modus, d.h.



Die neu angeordnete Liste wird wie folgt erzeugt:

← {} INTRO A z1 Q R A12 ENTER

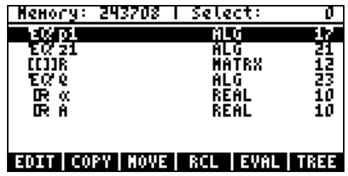
Anschließend geben Sie den Befehl ORDER, wie vorhin ein, d.h.
← PRG ▼ OK Wählen Sie MEMORY aus dem Programmiermenü
▼ ▼ ▼ ▼ OK Wählen Sie DIRECTORY aus dem Menü MEMORY
▲ ▲ OK Wählen Sie ORDER aus dem Menü DIRECTORY
Das Ergebnis ist die nachfolgende Anzeige:



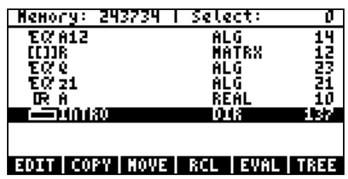
Verschieben von Variablen über das Menü FILES

Um eine Variable von einem Unterverzeichnis in ein anderes zu verschieben können wir das Menü FILES verwenden. So haben wir z.B. im Unterverzeichnis {HOME MANS INTRO} folgende Variablen *p1, z1, R, Q, A12*, α und *A*. Angenommen wir möchten die Variable *A12* ins Unterverzeichnis {HOME MANS} zu kopieren. Nachfolgend wie es gemacht wird: Drücken Sie ← FILES OK, um die Variablenliste anzuzeigen. Benutzen Sie die Pfeiltaste ▼, um die Variable *A12* auszuwählen und dann MOVE. Der Rechner wird sich mit der Anzeige 'PICK DESTINATION:' (Ziel wählen) melden Benutzen Sie die Pfeiltaste ▲, um das Unterverzeichnis

MANS auszuwählen und drücken dann OK. Nun sehen Sie den Inhalt des Unterverzeichnisses {HOME MANS INTRO}:



Beachten Sie, dass die Variable A12 nicht mehr da ist. Drücken Sie nun ← UPDIR , wird Ihnen der Inhalt des Unterverzeichnisses MANS, einschließlich der Variablen *A12* angezeigt.



Anmerkung: Mit Hilfe des Stacks können Sie eine Variable verschieben, indem Sie kopieren mit löschen einer Variable kombinieren. Wie Sie Variablen löschen können, wird im nächsten Abschnitt demonstriert.

Löschen von Variablen

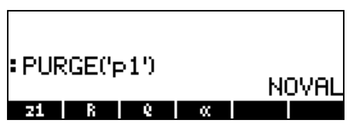
Variablen können mit Hilfe der Funktion PURGE gelöscht werden. Auf diese Funktion kann direkt mit Hilfe des Menüs TOOL (TOOL) oder FILES ← FILES OK zugegriffen werden

Verwenden des Befehls FILES

Der Befehl FILES kann dazu verwendet werden eine Variable einzeln zu löschen. Um eine Variable aus einem vorgegebenen Unterverzeichnis zu löschen, können wir das Menü FILES verwenden. So sind z.B. im Unterverzeichnis {HOME MANS INTRO} folgende Variablen *p1, z1, R, Q, α* und *A* verblieben.. Angenommen wir löschen die Variable *A*. Nachfolgend wie es gemacht wird: Drücken Sie ← FILES OK, um die Variablenliste anzuzeigen. Benutzen Sie die Pfeiltaste ▼ um die Variable A auszuwählen (es ist die letzte in der Auflistung), anschließend drücken Sie NXT PURGE YES. Nun sehen Sie den Inhalt des Unterverzeichnisses {HOME MANS INTRO} ohne die Variable A:

Anwenden der Funktion PURGE im Stack im algebraischen Modus

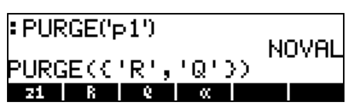
Wir beginnen wieder im Unterverzeichnis {HOME MANS INTRO}, in welchem die Variablen *p1, z1, Q, R* und α vorhanden sind. Wir wenden nun den PURGE Befehl an, um die Variable p1 zu löschen. Drücken Sie TOOL PURGE VAR p1 ENTER. In der Anzeige wird nun gemeldet, dass die Variable *p1* entfernt wurde:



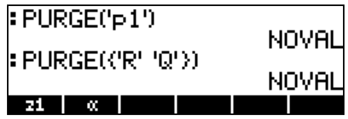
Mit dem PURGE Befehl können Sie mehr als eine Variable löschen, indem Sie deren Namen in die PURGE Argumentenliste eintragen. Wenn wir nun z.B. die Variablen R und Q gleichzeitig löschen möchten, können wir nachfolgende Übung versuchen. Drücken Sie:

TOOL PURGE ← {} ' VAR R ▶ → , ' VAR Q

In der Anzeige erscheint zu diesem Zeitpunkt folgender Befehl, der nun zur Ausführung bereitsteht:



Um den Löschvorgang der Variablen durchzuführen, drücken Sie ENTER. Die Anzeige weist nun nur noch die verbleibenden Variablen auf:



Anwenden der Funktion PURGE im Stack im RPN-Modus

Wir beginnen wieder im Unterverzeichnis {HOME MANS INTRO}, in welchem nun nur noch die Variablen *p1, z1, Q, R* und α vorhanden sind. Wir wenden nun den PURGE Befehl an, um die Variable p1 zu löschen. Drücken Sie ['] [p1] [ENTER] [TOOL] [PURGE]. In der Anzeige wird nun gemeldet, dass die Variable *p1* entfernt wurde:

Um zwei Variablen gleichzeitig zu löschen, sagen wir die Variablen *R* und *Q*, müssen wir zuerst eine Liste erstellen (im RPN-Modus müssen die Elemente der Liste nicht durch Komma, wie im algebraischen Modus, getrennt werden): [VAR] [←] {} ['] [R] [▶] ['] [Q] [ENTER]. Drücken Sie anschließend [TOOL] [PURGE] , um die Variablen zu löschen.

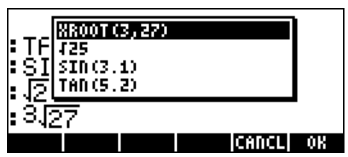
Die Funktionen UNDO und CMD

Die Funktionen UNDO und CMD sind dann nützlich, wenn Sie kürzlich gelöschte Befehle wiederherstellen möchten oder eine Operation rückgängig machen wollen, weil Ihnen ein Fehler unterlaufen ist. Diese Funktionen funktionieren in Zusammenhang mit der Taste HIST: UNDO bekommen Sie durch die Tastenfolge [→] UNDO , während Sie CMD über die Tastenfolge [←] CMD erhalten.

Um die Funktion UNDO zu veranschaulichen, versuchen Sie nachfolgende Übung im algebraischen (ALG) Modus: [5] [×] [4] [÷] [3] [ENTER]. Der Befehl UNDO ([→] UNDO) wird das Ergebnis einfach löschen. Die gleiche Übung im RPN-Modus, erfolgt über nachfolgende Tastenfolge: [5] [ENTER] [4] [ENTER] [×] [3] [ENTER] [÷]. Benutzen Sie [→] UNDO an dieser Stelle , wird die letzte Operation (20/3) rückgängig gemacht, während die ursprünglichen Dateien im Stack erhalten bleiben:

Um den Befehl CMD zu veranschaulichen, geben wir nachfolgendes im ALG Modus ein. Drücken Sie ENTER nach jeder Eingabe.

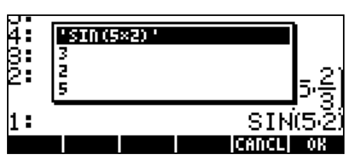
Als nächstes verwenden Sie die Funktion CMD (← CMD) um die letzten vier Befehle, die der Anwender eingegeben hat, anzuzeigen, d.h.



Mit Hilfe der Pfeiltasten (▲ ▼) können Sie durch diese Befehle nach oben und nach unten navigieren; dabei können Sie jeden hervorheben, den Sie neu eingeben möchten. Sobald Sie den Befehl ausgewählt haben, den Sie eingeben möchten, drücken Sie OK.
Die Funktion CMD funktioniert im RPN-Modus genauso, ausgenommen dass die Befehlsliste nur Zahlen oder Algebraiks auflisten wird. Die eingegebenen Funktionen werden nicht angezeigt. Versuchen Sie z.B. nachfolgendes Beispiel im RPN-Modus:

5 ENTER 2 ENTER 3 ÷ × SIN
' SIN 5 × 2 ENTER .

Drücken ← CMD erzeugt die nachfolgende Auswahlbox:

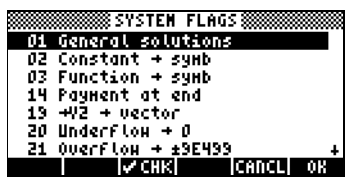


Wie Sie sehen können, werden die in der ersten Berechnung verwendeten Zahlen, 3, 2 und 5 in der Auswahlbox angezeigt, genauso die Algebraik 'SIN(5x2)', aber nicht die Funktion SIN, welche vorher in die Algebraik eingefügt wurde.

Flags

Ein Flag ist ein Boolscher Wert, welcher gesetzt oder gelöscht werden kann (wahr oder falsch), der eine bestimmte Einstellung des Rechners oder eine Option in einem Programm setzen kann. Flags, werden im Rechner über Zahlen identifiziert. Im Rechner existieren insgesamt 256 Flags, durchnumeriert zwischen -128 und 128. Die positiven Flags werden als Anwenderflags bezeichnet und diesem für Programmierzwecke zur Verfügung gestellt. Die durch negative Zahlen dargestellten Flags werden als Systemflags bezeichnet und beeinflussen die Arbeitsweise des Rechners.

Um sich die aktuellen Systemflageinstellungen anzusehen, drücken Sie die Taste MODE und anschließend die Funktionstaste FLAGS (d.h. F1). Sie erhalten eine Anzeige mit der Überschrift *SYSTEM FLAGS*, in welchem die Zahlen der Flags und die entsprechenden Einstellungen angezeigt werden.



(**Anmerkung:** In dieser Anzeige, da nur Systemflags vorhanden sind, wir nur der absolute Wert der Flag Zahl angegeben). Man sagt ein Flag ist gesetzt (*set*), wenn ein Häkchen (✓) vor der Flag-Zahl angezeigt wird. Andernfalls ist das Flag nicht *set (gesetzt)* oder *cleared (initialisiert)*. Um den Status eines System Flags zu ändern, drücken Sie die Funktionstaste CHECK! während das Flag, das Sie verändern möchten, hervorgehoben ist, oder Sie benutzen die +/- Taste . Sie können Sie Pfeiltasten (▲▼)benutzen, um in der Liste der Systemflags zu navigieren.

Obwohl es 128 System Flags gibt, werden nicht alle benutzt und einige davon gehören einfach nur zur internen Systemsteuerung. Systemflags, auf die der Anwender keinen Zugriff besitzt, werden auch nicht angezeigt. In Kapitel 24 wird eine komplette Liste der Flags vorgestellt.

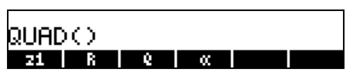
Beispiel einer Flageinstellung allgemeine Lösungen vs. Hauptwert

So z.B. ist der Standardwert für System Flag 01 *Allgemeine Lösungen* . Dies bedeutet, dass, wenn eine Gleichung mehrere Lösungen hat, werden alle Lösungen vom Rechner angezeigt, höchstwahrscheinlich in einer Liste . Drücken Sie die Funktionstaste CHECK können Sie das System Flag 01 auf Hauptwert setzen. Diese Einstellung zwingt den Rechner einen einzigen Wert anzuzeigen, auch als Hauptwert der Lösung bezeichnet.

Um dies in Aktion zu sehen, setzen Sie zuerst das System Flag 01 (d.h. wählen Sie *Principal Value (Hauptwert)*). Drücken Sie OK zweimal, um zur Normalanzeige des Rechners zurückzukehren. Versuchen wir eine quadratische Gleichung , sagen wir $t^2+5t+6 = 0$ mit dem Befehl QUAD zu lösen.

Algebraischer Modus

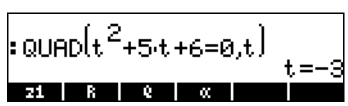
Benutzen Sie folgende Tastenfolge: ↱ CAT ALPHA Q . (Benutzen Sie dann die Pfeiltasten ▲▼, um QUAD auszuwählen) drücken Sie anschließend OK .



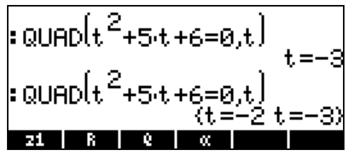
Um die Gleichung als erstes Argument der Funktion QUAD einzugeben, verwenden Sie die Tastenfolge:

↱ EQW ALPHA ↰ T Y^x 2 ▶ + 5 × ALPHA ↰ T + 6 ▲ ▲
↱ = 0 ENTER
↱ , ALPHA ↰ T ENTER

Die Lösung lautet:



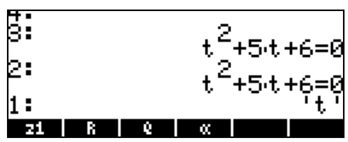
Ändern Sie nun die Einstellung von Flag 1 auf *General solutions (allgemeine Lösungen):* MODE FLAGS CHECK OK OK. Versuchen Sie eine weitere Lösung: ▲ ▲ ENTER ENTER. Im Ergebnis werden nun zwei Werte angezeigt:



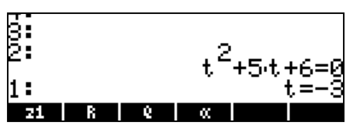
RPN-Modus

Erster Satz System Flags 01 (d.h. *Principal Value - Hauptwert*). Drücken Sie OK zweimal, um zur Normalanzeige des Rechners zurückzukehren. Anschließend tippen Sie die quadratische Gleichung wie folgt ein:

→ EQW ALPHA ← T yˣ 2 ▶ + 5 × ALPHA ← T + 6 ▲ ▲
→ = 0 ENTER
ENTER (behalten Sie eine zweite Kopie im RPN Stack)
' ALPHA ← T ENTER



Verwenden Sie nachfolgende Tastenfolge, um den QUAD Befehl zu starten: → CAT ALPHA Q. (Benutzen Sie dann die Pfeiltasten ▲▼, um QUAD auszuwählen) drücken Sie anschließend OK. In der Anzeige wird der Hauptwert angezeigt:



Ändern Sie nun die Einstellung von Flag 01 auf *General solutions (allgemeine Lösungen):* MODE FLAGS CHECK OK OK. Versuchen Sie eine weitere Lösung: ◀ ' ALPHA ← T ENTER → CAT ALPHA Q (Benutzen Sie dann die Pfeiltasten ▲▼, um den Befehl QUAD auszuwählen) drücken Sie anschließend OK . In der Anzeige bekommen Sie die nachfolgenden zwei Lösungen:

Weitere erwähnenswerte Flags

Nehmen wir nochmals die aktuelle Flag Einstellung, indem wir die Taste MODE drücken und anschließend die Funktionstaste FLAGS . Stellen Sie sicher, dass Sie das System Flag 01, welches in einer früheren Übung gesetzt wurde, bereinigen. Verwenden Sie die Pfeiltasten (▲▼), um in der System Flag Liste zu navigieren.

Einige interessante Flag und derer bervorzugter Wert für die Übungen aus diesem Handbuch sind:

02 Constant → symb : Konstante Werte (z.B. π) werden als Symbole beibehalten

03 Function → symb : Funktionen werden nicht automatisch ausgewertet, statt dessen werden diese als symbolische Ausdrücke geladen.

*27 'X+Y*i' → (X,Y)* : Komplexe Zahlen werden als geordnete Paare dargestellt

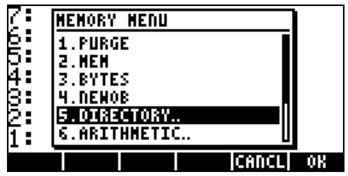
60 [α][α] Datensperren : Die Tastenfolge ALPHA ALPHA stellt die alphatesische Tastatur festDrücken Sie OK zweimal, um zur Normalanzeige des Rechners zurückzukehren.

CHOOSE Kästchen vs. Funktions-MENU

In einigen Beispielen dieses Kapitels, konnten wir Menü-Befehlslisten auf dem Display angezeigt, sehen. Diese Menülisten werden als *CHOOSE* Kästchen bezeichnet. Um z.B. mit dem Befehl ORDER die Variablen in einem Verzeichnis neu anzuordnen, die wir verwendet haben

← PRG ▼ Zeige Menüliste PROG und wähle MEMORY

OK ▼ ▼ ▼ ▼

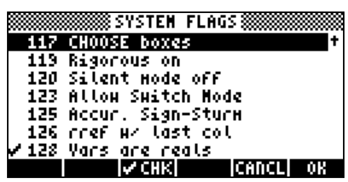


OK ▲ ▲ Zeige Menüliste DIRECTORY und wähle ORDER

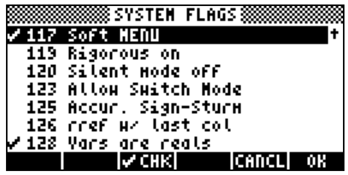
OK starten Sie den Befehl ORDEREs gibt eine Alternative diese Menüs über die Funktionstasten zu erreichen, und zwar durch setzen des System-Flags 117. Um dieses Flag zu setzen versuchen Sie folgendes:

MODE FLAGS ▲ ▲ ▲ ▲ ▲ ▲ ▲

In der Anzeige erscheint Flag 117 nicht gesetzt (CHOOSE Kästchen), wie nachfolgend zu sehen ist:



Drücken Sie die Funktionstaste CHECK um Flag 117 im Funktions-MENU zu setzen. In der Anzeige wird diese Änderung angezeigt:

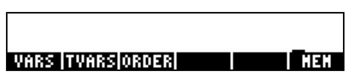


Drücken Sie zweimal, um zur Normalanzeige des Rechners zurückzukehren. Nun werden wir versuchen den Befehl ORDER mit ähnlicher Tastenfolge wie oben zu finden, d.h. wir starten mit ← PRG .
Beachten Sie, dass wir in diesem Fall anstelle einer Menüliste, Funktionstasten für das Menü mit den verschiedenen Optionen für das Menü PROG erhalten, d.h.

Drücken Sie F2 , um das Funktionsmenü MEMORY (MEM) auszuwählen. In der Anzeige erscheint nun:

Drücken Sie F5 , um das Funktionsmenü DIRECTORY (DIR) auszuwählen.

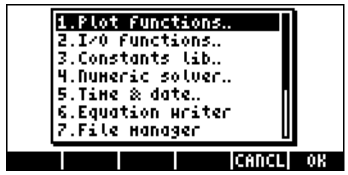
Der Befehl ORDER wird nicht angezeigt. Um diesen anzuzeigen benutzen wir die Taste NXT :



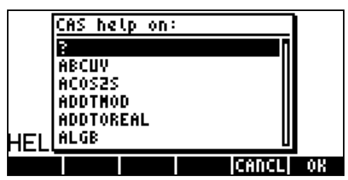
Um den Befehl ORDER zu starten drücken wir die Funktionstaste F3 (ORDER). Obwohl nicht auf ein bestimmtes Beispiel angewendet, zeigt diese Übung zwei Optionen für Menüs im Rechner an (CHOOSE Kästchen und Funktions-MENUs).

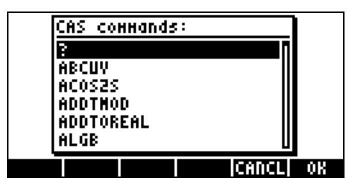
Ausgwählte CHOOSE Kästchen

Einige Menüs erzeugen nur CHOOSE Kätschen, so z.B.

- Das Menü APPS (APPlicationS - Anwendungen), gestartet mit der Taste APPS, erste Taste in der zweiten Reihe von oben:
- 

- Das Menü CAT (CATalog - Katalog), gestartet mit der Taste ↱ CAT, zweite Taste in der vierten Reihe von oben:
- 

- Das Menü HELP, gestartet mit TOOL NXT HELP
- 

- Das Menü CMDS (CoMmanDS – Befehle), welches innerhalb des EquationWriters mit der Tastenfolge ↱ EQW NXT CMDS aktiviert wird
- 

Kapitel 3
Berechnung mit reellen Zahlen

In diesem Kapitel wird die Benutzung des Rechners für Operationen und Funktionen in Zusammenhang mit reellen Zahlen erläutert. Die hier aufgeführten Operationen, werden in den meisten Berechnungen in der Physik und in der Technik angewendet. Der Benutzer sollte sich mit der Tastatur insofern auskennen, um bestimmte, auf der Tastatur vorhandene Funktionen aufzufinden (z.B. SIN, COS, TAN, usw.) Es wird angenommen, dass der Benutzer weiß wie er den Rechner bedient, d.h. den Betriebsmodus auswählt (siehe Kapitel 1), Menüs und Auswahlkästchen (Kapitel 1) verwendet und wie man mit Variablen umgeht (Kapitel 2).

Überprüfen der Einstellungen des Rechners

Um die aktuellen CAS Einstellungen des Rechners zu überprüfen, beachten Sie die oberste Zeile im Display des Rechners, im Normalbetrieb. So könnten Sie z.B. die folgenden Einstellungen sehen: RAD XYZ DEC **R** = 'X'

So steht RADiane für winkelförmige Maße, XYZ für rechtwinklige (Kartesische) Koordinaten, DECimal Zahlenbasis, **R**eelle Zahlen bevorzugt = "genaue" Ergebenisse und 'X' ist der Wert der unabhängigen Standardvariablen.

Eine weitere Möglichkeit diese Optionen anzuzeigen DEG R∠Z HEX C ~ 't'

So DEGrees (Grade) als Winkelmaß, R∠Z für Polar-Koordinaten, HEXadezimalzahlen, komplexe Zahlen erlaubt, ~ steht für "ungefähre" Ergebenisse und 't' als die unabhängige Standardvariable.

Im allgemeinen sind auf diesem Teil des Displays sieben Elemente enthalten. Jedes Element wird durch eine Zahl zwischen 1 bis 7 identifiziert. Die für jedes Element möglichen Werte, werden zwischen den Klammern nach der Beschreibung des Elementes angegeben. Ebenso wird die Erklärung für jeden einzelnen dieser Werte angezeigt:

1. Spezifikation des Winkelmaßes (DEG, RAD, GRD)
   DEG: Grade, 360 Grad bilden einen kompletten Kreis

RAD: Radiane, 2π Radiane bilden einen kompletten Kreis
GRD: Zentesimalgrade, 400 Zentesimalgrade bilden einen kompletten Kreis

2. Spezifikationen des Koordinatensystems (XYZ, R∠Z, R∠∠). Das Symbol ∠ steht für Winkelkoordinaten.
   XYZ: Kartesisch oder rechtwinklig (x,y,z)
   R∠Z: zylindrische Polarkoordinaten (r,θ,z)
   R∠∠: sphärische Koordinaten (ρ,θ,φ)
3. Zahlenbasen Spezifikation (HEX, DEC, OCT, BIN)
   HEX: Hexadezimalzahlen (Basis 16)
   DEC: Dezimalzahlen (Basis 10)
   OCT: Oktalzahlen (Basis 8)
   BIN: Binärzahlen (Basis 2)
4. Reeller oder komplexer Modus (**R, C**)
   R: reelle Zahlen
   C: komplexe Zahlen
5. Realmodus oder Annäherungsmodus (=,~)
   = exakter (symbolischer) Modus
   ~ Annäherungsmodus (numerischer)
6. unabhängige Standard CAS Variable (z.B. 'X', 't', usw.)

Überprüfen des Rechner Modus

Im RPN-Modus werden die verschiedenen Stack Ebenen auf der linken Seite des Displays angezeigt. Im ALGEBRAISCHEN Modus gibt es keine durchnummerierten Stack Ebenen, und das Wort ALG wird in der oberen Zeile des Displays rechts angezeigt. Der Unterschied zwischen diesen beiden Operationsmodi wurde im Detail in Kapitel 1 beschrieben.

Berechnungen mit reellen Zahlen

Um Berechnungen mit reellen Zahlen durchzuführen ist es ratsam das CAS auf *Real* (im Gegensatz zum *Complex* Modus) eingestellt zu haben. In manchen Fällen könnten Sie ein komplexes Ergebnis erhalten, wobei Sie vom Rechner aufgefordert werden in den *Complex* Modus umzuschalten. Der *Exact* Modus ist der Standardmodus für die meisten Berechnungen. Deshalb sollten Sie mit Ihren Berechnungen in diesem Modus starten. Sollte es erforderlich sein in den *Approx* Modus umzuschalten, werden Sie vom Rechner dazu aufgefordert dies zu tun. Es gibt keine bevorzugte Auswahl für das Winkelmaß oder für die Zahlenbasisspezifikation. Berechnungen mit reellen Zahlen werden sowohl im algebraischen (ALG) Modus als auch im Reverse Polish Notation (RPN) Modus dargestellt.

Änderung des Vorzeichens einer Zahl, Variablen oder eines Ausdrucks

Drücken Sie die Taste +/-. Im ALG-Modus können Sie, bevor Sie eine Zahl eingeben +/- drücken z.B. +/- 2 • 5 ENTER. Das Ergebnis = -2,5. Im RPN-Modus, müssen Sie vorerst mindestens einen Teil der Zahl eingeben und dann die Taste +/- verwenden, z.B. 2 • 5 +/-. Das Ergebnis = -2,5. Benutzen Sie die Funktion +/-, wenn keine Befehlszeile vorhanden ist, wird der Rechner die Funktion NEG (Umkehr des Vorzeichens) auf das Objekt in der ersten Stack Ebene anwenden.

Die Umkehrfunktion

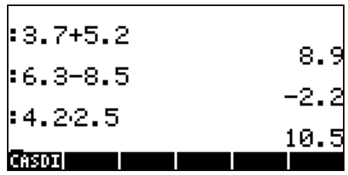
Verwenden Sie die Taste 1/x. Im ALG-Modus, drücken Sie zuerst 1/x , gefolgt von einer Zahl oder einem algebraischen Ausdruck, z.B. 1/x 2. Das Ergebnis = ½ oder 0,5. Im RPN-Modus geben Sie zuerst die Zahl, dann die Funktion ein, z.B.: 4 ENTER 1/x. Das Ergebnis = ¼ oder 0,25.

Addition, Subtraktion, Multiplikation, Division

Verwenden Sie die reinen Operationstasten, und zwar + − × ÷. Im ALG-Modus, drücken Sie einen Operanden, dann einen Operator, dann wiederum einen Operanden gefolgt von einem ENTER, um ein Ergebnis zu erzielen. Beispiele:

3 • 7 + 5 • 2 ENTER
6 • 3 − 8 • 5 ENTER
4 • 2 × 2 • 5 ENTER
2 • 3 ÷ 4 • 5 ENTER

Die ersten drei Operationen werden in obiger Anzeige dargestellt:



Im RPN-Modus geben Sie einen Operanden nach dem anderen, jeweils durch ein [ENTER] getrennt, ein, dann drücken Sie die Taste für den Operator. Beispiele:

[3] [.] [7] [ENTER] [5] [.] [2] [+]
[6] [.] [3] [ENTER] [8] [.] [5] [−]
[4] [.] [2] [ENTER] [2] [.] [5] [×]
[2] [.] [3] [ENTER] [4] [.] [5] [÷]

Im RPN-Modus können Sie alternativ dazu die Operanden mit einem Leerzeichen ([SPC]) trennen, bevor Sie die Befehlstaste drücken. Beispiele:

[3] [.] [7] [SPC] [5] [.] [2] [+]
[6] [.] [3] [SPC] [8] [.] [5] [−]
[4] [.] [2] [SPC] [2] [.] [5] [×]
[2] [.] [3] [SPC] [4] [.] [5] [÷]

Verwendung von Klammern

Klammern können dazu verwendet werden, Operationen zu gruppieren, oder aber auch um Funktions-Argumente einzuschließen. Klammern erhält man über die Tastenkombination [←] [()] . Klammern kommen immer paarweise. So z.B. zur Berechnung von (5+3.2)/(7-2.2):

Im ALG-Modus:

[←] [()] [5] [+] [3] [.] [2] [▶] [÷] [←] [()] [7] [−] [2] [.] [2] [ENTER]

Im RPN-Modus benötigen Sie keine Klammern, die Berechnung erfolgt direkt im Stack:

[5] [ENTER] [3] [.] [2] [ENTER] [+] [7] [ENTER] [2] [.] [2] [ENTER] [−] [÷]

Wenn Sie im RPN-Modus den Ausdruck in Anführungszeichen schreiben, können Sie diesen wie im algebraischen Modus eingeben.

' ↰ () 5 + 3 • 2 ▶ ÷
↰ () 7 − 2 • 2 ENTER EVAL

In beiden Fällen, im ALG- wie auch im RPN-Modus, kann der EquationWriter dazu benutzt werden:

↱ EQW 5 + 3 • 2 ▶ ÷ 7 − 2 • 2

Der Ausdruck kann innerhalb des EquationWriters ausgewertet werden, indem Sie nachfolgende Tastenfolgen benutzen

▲ ▲ ▲ ▲ EVAL or, ↱ ▲ EVAL

Funktion Absoluter Wert

Die Funktion absoluter Wert, ABS, kann über die Tastenkombination ↰ ABS aufgerufen werden. Sollten Sie im Stack im ALG-Modus berechnen, müssen Sie die Funktion vor dem Argument eingeben, so z.B. :

↰ ABS +/- 2 • 3 2 ENTER

Im RPN-Modus geben Sie zuerst die Zahl, dann die Funktion ein, z.B.:

2 • 3 2 +/- ↰ ABS

Quadrate und Quadratwurzeln

Die Quadratfunktion, SQ, kann über die Tastenkombination ↰ x^2 aufgerufen werden. Sollten Sie im Stack im ALG-Modus berechnen, müssen Sie die Funktion vor dem Argument eingeben, so z.B. :

↰ x^2 +/- 2 • 3 ENTER

Im RPN-Modus geben Sie zuerst die Zahl, dann die Funktion ein, z.B.:

2 • 3 +/- ↰ x^2

Die Quadratwurzelfunktion, √, kann über die Taste R aufgerufen werden. Sollten Sie im Stack im ALG-Modus berechnen, müssen Sie die Funktion vor dem Argument eingeben, so z.B. :

√x 1 2 3 • 4 ENTER

Im RPN-Modus geben Sie zuerst die Zahl, dann die Funktion ein, z.B.:

1 2 3 • 4 √x

Potenzen und Wurzeln

Die Potenzfunktion, ^, wird über die Taste Y^x aufgerufen. Wenn Sie im Stack im ALG-Modus berechnen, geben Sie die Base (y) gefolgt von der Taste Y^x und anschließend den Exponenten (x) ein, z.B.

5 • 2 Y^x 1 • 2 5

Im RPN-Modus geben Sie zuerst die Zahl, dann die Funktion ein, z.B.:

5 • 2 ENTER 1 • 2 5 ENTER Y^x

Die Wurzelfunktion XROOT*(y,x)* kann über die Tastenkombination ↱ $\sqrt[x]{y}$ erreicht werden. Sollten Sie im Stack im ALG-Modus berechnen, müssen Sie die Funktion XROOT, gefolgt von den Argumenten (*y,x*), durch Komma getrennt, eingeben, so z.B. :

↱ $\sqrt[x]{y}$ 3 ↱ , 2 7 ENTER

Im RPN-Modus müssen Sie zuerst das Argument *y*, dann *x* und schließlich dann die Funktion aufrufen, so z.B. 2 7 ENTER 3 ENTER ↱ $\sqrt[x]{y}$

Logarithmen mit der Basis 10 und Zehnerpotenzen

Die Berechnung von Logarithmen mit der Basis 10 werden über die Tastenkombination ↱ LOG (Funktion LOG) berechnet, während die Umkehrfunktion (ALOG oder Algorhitmus) über die Tastenkombination ↰ 10^x berechnet wird. Im ALG-Modus wird die Funktion vor dem Argument eingegeben:

↱ LOG 2 • 4 5 ENTER
↰ 10^x +/- 2 • 3 ENTER

Im RPN-Modus wird das Argument vor der Funktion eingegeben:

2 . 4 5 ENTER ↱ LOG
2 . 3 +/- ENTER ↰ 10^x

Verwendung von Zehnerpotenzen bei der Dateneingabe

Zehnerpotenzen, d.h. Zahlen wie -4.5×10^{-2}, usw., werden mit Hilfe der Taste EEX eingegeben. So z.B. im ALG-Modus:

+/- 4 . 5 EEX +/- 2 ENTER

Oder im RPN-Modus:

4 . 5 +/- EEX 2 +/- ENTER

Natürliche Logarithmen und Exponentialfunktionen

Natürliche Logarithmen (d.h. Logarithmen mit der Basis *e = 2,7182818282*) werden über die Tastenkombination ↱ LN (Funktion LN) berechnet, während ihre Umkehrfunktion, die Exponentialfunktion (Funktion EXP) unter Verwendung von ↰ e^x berechnet werden. Im ALG-Modus wird die Funktion vor dem Argument eingegeben:

↱ LN 2 . 4 5 ENTER
↰ e^x +/- 2 . 3 ENTER

Im RPN-Modus wird das Argument vor der Funktion eingegeben:

2 . 4 5 ENTER ↱ LN
2 . 3 +/- ENTER ↰ e^x

Trigonometrische Funktionen

Drei trigonometrische Funktionen sind bereits über die Tastatur abrufbar: Sinus (SIN), Cosinus (COS), und Tangens (TAN). Die Argumente dieser Funktion sind Winkel, deshalb können diese in jedes beliebige Winkelsystem (Grade, Radiane, Zentesimalgrade) eingegeben werden. So z.B. können wir nachfolgende trigonometrische Funktionen mit ausgewählter DEG (Grad) Funktion berechnet werden:

Im ALG-Modus:

SIN 3 0 ENTER
COS 4 5 ENTER
TAN 1 3 5 ENTER

Im RPN-Modus:

3 0 ENTER SIN
4 5 ENTER COS
1 3 5 ENTER TAN

Inverse trigonometrische Funktionen

Die inversen trigonometrischen Funktionen stehen über die Tastatur zur Verfüfung und sind Arcsinus (ASIN), Arccosinus (ACOS) und Arctangens (ATAN) über die jeweiligen Tastenkombinationen ← ASIN , ← ACOS und ← ATAN . Da die Inversen der trigonometrischen Funktionen Winkel darstellen, werden die Ergebnisse in den ausgewählten Winkelmaßen (DEG, RAD, GRD) ausgegeben. Nachfolgend einige Beispiele:

Im ALG-Modus:

← ASIN 0 • 2 5 ENTER
← ACOS 0 • 8 5 ENTER
← ATAN 1 • 3 5 ENTER

Im RPN-Modus:

0 • 2 5 ENTER ← ASIN
0 • 8 5 ENTER ← ACOS
1 • 3 5 ENTER ← ATAN

Alle oben aufgeführten Funktionen und zwar ABS, ABS, SQ, √, ^, XROOT, LOG, ALOG, LN, EXP, SIN, COS, TAN, ASIN, ACOS, ATAN, können mit den Grundrechenarten (+ − × ÷) zur Bildung komplexerer Ausdrücke kombiniert werden. Der EquationWriter, dessen Operationen in Kapitel 2 beschrieben wurde, ist genau das Richtige für derartige Ausdrücke, egal in welchem Modus der Rechner eingestellt ist.

Unterschied zwischen Funktionen und Operatoren

Funktionen wie ABS, ABS, SQ, √, LOG, ALOG, LN, EXP, SIN, COS, TAN, ASIN, ACOS, ATAN benötigen ein einziges Argument. Deshalb ist deren Anwendung im ALG-Modus recht einfach, z.B. ABS(x). Einige Funktionen wie XROOT benötigen zwei Argumente, z.B. XROOT(x,y). Diese Funktion hat die entsprechende Tastenfolge [→] [ˣ√y] .

Operatoren hingegen, werden nach einem einzigen Argument oder zwischen zwei Argumenten eingesetzt. Der faktorielle Operator (!) z.B. wird nach einer Zahl eingesetzt, z.B. [5] [ALPHA] [→] [2] [ENTER]. Da dieser Operator ein einziges Argument benötigt, wird dieser als monadisch bezeichnet. Operatoren, welche zwei Argumente benötigen, wie z.B. [+] [−] [×] [÷] [Y^x] werden als Binäroperatoren bezeichnet, z.B. [3] [×] [5], oder [4] [Y^x] [2].

Funktionen von reellen Zahlen im Menü MTH

Das Menü MTH (MaTHematics) beinhaltet eine Reihe von mathematischen Funktionen die hauptsächlich auf reelle Zahlen angewandt werden. Um in dieses Menü zu gelangen drücken Sie die Tastenkombination [←] [MTH] . Mit der Standardeinstellung auf CHOOSE Kästchen für System Flag 117 (siehe Kapitel 2), wird das Menü MTH als Menüliste dargestellt:

Da im Rechner eine Vielzahl an mathematischen Funktionen vorhanden sind, wurde das MTH-Menü nach Objekttyp der Funktion, der diese zugeordnet sind, sortiert. So z.B. sind die Optionen 1. *VECTOR.., 2. MATRIX.* und *3. LIST..* eben denselben Datentypen zugeordnet (d.h. Vektoren, Matrizen und Listen), welche im Detail in einem späteren Kapitel erörtert werden. Die Optionen *4. HYPERBOLIC..* and *5. REAL..* sind reellen Zahlen zugeordnet und werden in diesem Kapitel erörtert. Option *6. BASE..* wird für Umrechnungen von Zahlen in unterschiedlichen Basen verwendet und werden ebenfalls in einem späteren Kapitel erörtert. Option *7. PROBABILITY..* wird für Wahrscheinlichkeits-Anwendungen eingesetzt und in einem der nächsten Kapitel erörtert. Option *8. FFT..* (Fast Fourier Transform) ist eine Anwendung zur Signalbearbeitung und wird in einem anderen Kapitel erörtert. Option *9. COMPLEX..* enthält Funktionen für komplexe Zahlen, welche im nächsten

Kapitel erörtert werden. Option *10. CONSTANTS* ermöglicht den Zugang zu den Konstanten im Rechner. Diese Option wird weiter unten in diesem Abschnitt erörtert. Schließlich die Option *11. SPECIAL FUNCTIONS..* schließt Funktionen für höhere Mathematik ein, welche auch in diesem Abschnitt erörtert werden.

Im allgemeinen, um eine dieser Funktionen anzuwenden, sollten Sie auf die Anzahl und Anordnung, der für jede Funktion erforderlichen Argumente achten und nicht vergessen, dass im ALG-Modus immer zuerst die Funktion und dann das Argument eingegeben wird, während im RPN-Modus erst das Argument in den Stack eingegeben und anschließend die Funktion ausgewählt wird.

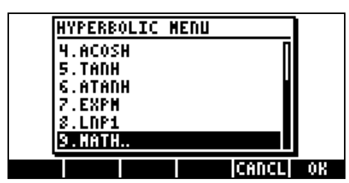
Verwendung der Rechenmenüs:

1. Da die Funktionsweise von MTH Funktionen (und viele andere Rechnermenüs) sehr ähnlich ist, werden wir nur die Funktionsweise der Menü-Option *4. HYPERBOLIC..* in diesem Abschnitt beschreiben, mit der Absicht eine generelle Funktionsweise der einzelnen Rechnermenüs aufzuzeigen. Achten Sie besonders auf die Vorgehensweise bei der Auswahl verschiedener Optionen.

2. Für eine schnelle Auswahl der nummerierten Optionen der Menüliste (oder CHOOSE Kästchen), geben Sie einfach die entsprechende Nummer der gewünschten Option in die Tastatur. Um z.B. die Option *4. HYPERBOLIC..* im Menü MTH auszuwählen, drücken Sie einfach die 4.

Hyperbolische Funktionen und deren Inverse

Um das hyperbolische Funktionsmenü aufzurufen wählen Sie im MTH Menü die Option *4. HYPERBOLIC..* und betätigen Sie anschließend die Taste OK



Die hyperbolischen Funktionen sind:

Hyperbolische Sinusfunktion, SINH und deren Inverse ASINH oder $sinh^{-1}$
Hyperbolische Cosinusfunktion, COSH und deren Inverse ACOSH oder $cosh^{-1}$
Hyperbolische Tangensfunktion, TANH und deren Inverse ATANH oder $tanh^{-1}$

Dieses Menü enthält zusätzlich die nachfolgenden Funktionen:

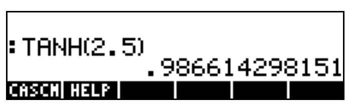
$$EXPM(x) = \exp(x) - 1,$$
$$LNP1(x) = \ln(x+1).$$

Schließlich dann Option *9. MATH,* welches den Anwender zurück ins Menü MTH bringt.

So z.B. zum berechnen der Funktion *tanh(2,5)*, benötigen Sie im ALG-Modus folgende Tastenfolge:

← MTH	Auswahl *MTH* Menü
4 OK	Auswahl Menü *4. HYPERBOLIC..*
5 OK	Auswahl der Funktion *5. TANH*
2 . 5 ENTER	Berechnen von *tanh(2,5)*

In der Anzeige erscheint folgende Ausgabe:



Für die gleiche Kalkulation im RPN-Modus wird nachfolgende Tastenfolge benötigt:

2 . 5 ENTER	Geben Sie das Argument in den Stack
← MTH	Auswahl *MTH* Menü
4 OK	Auswahl Menü *4. HYPERBOLIC..*
5 OK	Auswahl der Funktion *5. TANH*

Die Lösung lautet:

Die aufgeführten Operationen setzen voraus, dass Sie die Standard Einstellungen für System Flag 117 (*CHOOSE* Kästchen) benutzen: Haben Sie die Einstellungen dieses Flags auf *SOFT menu* (siehe Kapitel 2) eingestellt, wird das Menü MTH wie folgt aussehen (linke Seite ALG-Modus, rechte Seite RPN-Modus):

Wenn Sie die Taste NXT drücken, erscheinen die weiteren noch zur Verfügung stehenden Optionen:

Anmerkung: Durch drücken von PREV kommen Sie zu den ersten Optionen des *MTH* Menüs zurück. Mit der Tastenkombination erhalten Sie eine Auflistung der Menüfunktionen in der Anzeige, so z.B.

So, um z.B. das hyperbolische Funktionsmenü aus diesem Menü aufzurufen, drücken Sie die Taste HYP, um nachfolgende Darstellung zu erhalten:

Schließlich, um die hyperbolische Funktion Tangens (atanh) zu erhalten, drücken Sie einfach die Taste ATANH.

Anmerkung: Um zusätzliche Optionen dieses Funktionsmenüs anzuzeigen, drücken Sie entweder die Taste NXT oder die Tastenkombination ← PREV .

Um z.B. dieselbe Funktion *tanh(2,5),* im ALG-Modus zu berechnen, wenn auf SOFT Menü eingestellt über *CHOOSE* Kästchen, gehen Sie wie folgt vor:

← MTH	Auswahl *MTH* Menü
HYP	Wählen Sie das Menü *HYPERBOLIC..*
TANH	Auswahl der Funktion *TANH*
2 . 5 ENTER	Berechnen von *tanh(2,5)*

Denselben Wert errechnen Sie im RPN-Modus über nachfolgende Tastenfolge:

2 . 5 ENTER	Geben Sie das Argument in den Stack
← MTH	Auswahl *MTH* Menü
HYP	Wählen Sie das Menü *HYPERBOLIC..*
TANH	Auswahl der Funktion *TANH*

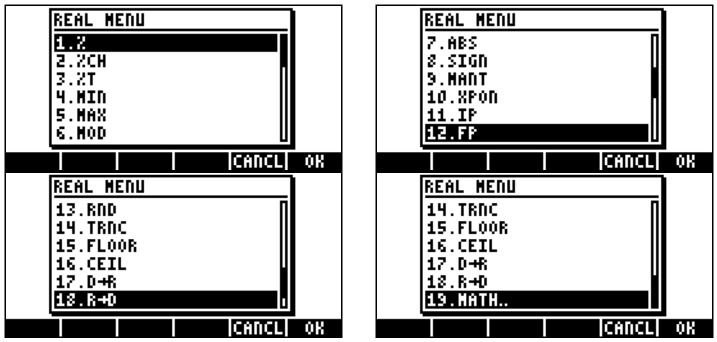
Als Beispiel für Anwendungen von hyperbolischen Funktionen, überprüfen Sie die nachfolgenden Werte:

sinh (2,5) = 6,05020..	$sinh^{-1}(2,0) = 1,4436…$
cosh (2,5) = 6,13228..	$acosh^{-1}(2,0) = 1,3169…$
tanh (2,5) = 0,98661..	$tanh^{-1}(0,2) = 0,2027…$
expm(2,0) = 6,38905….	*lnp1(1) = 0,69314….*

Um es nochmals klar zu stellen, die allgemeine, in diesem Abschnitt beschriebene Prozedur, ist für alle Auswahlfunktionen in jedem Menü des Rechners anwendbar.

Funktionen zu reellen Zahlen

Bei der Auswahl der Option *5. REAL..* aus dem Menü MTH, mit System Flag 117 auf *CHOOSE* Kästchen gesetzt, wird folgende Menüliste erzeugt.



Option *19. MATH, bringt* den Anwender zurück ins Menü MTH. Die übriggebliebenen Funktionen werden in sechs verschiedene Gruppen angeordnet und werden nachfolgend beschrieben.

Wenn das System Flag 117 auf SOFT Menüs gesetzt ist, sehen die Funktionen *REAL* im ALG-Modus wie folgt aus (ALG-Modus wird verwendet, die gleichen Funktionstasten stehen aber auch im RPN-Modus zur Verfügung):

Die allerletzte Funktion, MTH, bringt den Anwender zurück ins Menü *MTH*.

Prozentfunktionen

Diese Funktionen dienen der Prozentrechnung und gleichartiger Werte wie folgt:

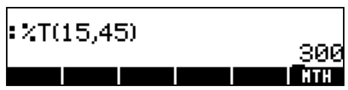
% (y,x) : berechnet den Prozentsatz x von y

%CH(y,x) : berechnet 100(y-x)/x, d.h. den Prozentsatzwechsel

%T(y,x) : berechnet 100 x/yDiese Funktionen benötigen zwei Argumente, wir veranschaulichen nachfolgend die Berechnung *%T(15,45)*, d.h., Berechnung 15% von 45. Nehmen wir an, der Rechner ist im ALG-Modus und System Flag 117 auf *CHOOSE* Kästchen gesetzt. Das Verfahren sieht wie folgt aus:

← MTH	Auswahl *MTH* Menü
5 OK	Auswahl Menü *5. REAL..*
3 OK	Wählen Sie die 5, Funktion *%T*
1 5	Geben Sie das erste Argument ein
→ ,	Geben Sie ein Komma ein, um die Argumente voneinander zu trennen
4 5	Geben Sie das zweite Argument ein
ENTER	Berechnen Sie die Funktion

Nachfolgend das Ergebnis:



Im RPN-Modus befindet sich Argument *y*, in der zweiten Stack Ebene, während sich das Argument *x* in der ersten Stack Ebene befindet. Das bedeutet, Sie sollten *x* vor *y* nur im ALG-Modus eingeben. Somit erfolgt die Berechnung von von *%T(15,45)* im RPN-Modus, mit System Flag 117 auf *CHOOSE* Kästchen gesetzt, wie folgt:

1 5 ENTER	Geben Sie das erste Argument ein
4 5 ENTER	Geben Sie das zweite Argument ein
← MTH	Auswahl *MTH* Menü
5 OK	Auswahl Menü *5. REAL..*
3 OK	Wählen Sie die 5, Funktion *%T*

Anmerkung: Die Beispiele in diesem Abschnitt veranschaulichen im Allgmeinen den Einsatz von Funktionen mit zwei Argumenten. Funktionen mit 3 oder mehr Argumenten, können aus diesen Beispielen generalisiert werden.

Als Beispiel für Anwendungen von prozentbezogenen Funktionen, überprüfen Sie die nachfolgenden Werte: *%(5,20) = 1, %CH(22,25) = 13.6363.., %T(500,20) = 20*

Minimum und Maximum

Verwenden Sie diese Funktionen, um den Minimal- oder Maximalwert von zwei Argumenten zu berechnen.

MIN(x,y) : Minimalwert von x und y

MIN(x,y) : Minimalwert von x und y

Als Beispiel, überprüfen Sie, ob *MIN(-2,2) = -2, MAX(-2,2) = 2*

Modulo:

MOD: y mod x = Rest von y/x, d.h. wenn x und y Integerzahlen sind, y/x = d + r/x, wobei d = Quotient, r = Rest. In diesem Fall, r = y mod x.

Beachten Sie, dass MOD keine Funktion darstellt, sondern eher einen Operator, d.h. im ALG-Modus, sollte MOD als `y MOD x`, und nicht als `MOD(y,x)` verwendet werden. Somit ist die Vorgehensweise von MOD ähnlich von [+], [−], [×], [÷].

Als Beispiel, überprüfen Sie, ob *15 MOD 4 = 15 mod 4 = Rest von 15/4 = 3,* ist.

Absoluter Wert, Vorzeichen, Mantisse, Exponent, Integer und Brüche

ABS(x) : berechnet den absoluten Wert |x|

SIGN(x): legt das Vorzeichen von x fest, d.h. -1, 0 oder 1

MANT(x): bestimmt die Mantisse einer Zahl basierend auf $\log_{10}$.

XPON(x): bestimmt die Zehnerpotenz in der Zahl

IP(x) : bestimmt den Integerteil einer reellen Zahl

FP(x) : bestimmt den Bruchteil einer reellen Zahl

Als Übung überprüfen Sie, ob *ABS(-3) = |-3| = 3, SIGN(-5) = -1, MANT(2540) = 2,540, XPON(2540) = 3, IP(2,35) = 2, FP(2,35) = 0,35.*

Aufrunden, abschneiden nächst kleinere *(floor)* und nächst höhere (*ceiling)* Grenzzahl Funktion

RND(x,y) : rundet y auf x Dezimalstellen auf

TRNC(x,y) : schneidet y auf x Dezimalstellen ab

FLOOR(x) : größtmögliche Integerzahl, kleiner oder gleich x

CEIL(x) : kleinstmögliche Integerzahl, größer oder gleich x.

Als Übung überprüfen sie, ob *RND(1,4567,2) = 1.46, TRNC(1,4567,2) = 1,45, FLOOR(2,3) = 2, CEIL(2,3) = 3*

Radian-in-Grad und Grad-in-Radian Funktionen

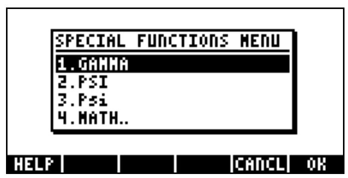
D→R (x) : konvertiert Grade in Radiane

R→D (x) : konvertiert Radiane in Grade um.

Als Übung überprüfen Sie, ob *D→R(45) = 0,78539* (d.h., 45° = $0{,}78539^{\text{rad}}$), *R→D(1,5) = 85,943669..* (d.h., $1{,}5^{\text{rad}}$ = 85,943669..°).

Sonderfunktionen

Option *11. Special functions…* (Sonderfunktionen) im MTH Menü beinhaltet folgende Funktionen:



GAMMA: Die Gammafunktion $\Gamma(\alpha)$

PSI: N-te Ableitung der Digamma-Funktion

Psi: Digamma-Funktion, Ableitung der ln(Gamma)

Die Gamma-Funktion wird wie folgt definiert $\Gamma(\alpha) = \int_0^\infty x^{\alpha-1} e^{-x} dx$. Diese Funktion wird in der angewandten Mathematik für Wissenschaft und Technik, wie auch in Wahrscheinlichkeits – und Statistik-Rechnungen eingesetzt.

Faktorielle einer Zahl

Die Faktorielle einer positiven Integerzahl n wird als *n!=n·(n-1)·(n-2) …3·2·1*, mit *0! = 1* definiert. Mit der Tastenfolge [ALPHA][↱][2] ist die Funktion Faktorielle im Rechner verfügbar. In beiden Modi, dem ALG und dem RPN, geben Sie zuerst die Zahl, gefolgt von der Tastenfolge [ALPHA][↱][2] ein. Beispiel: [5][ALPHA][↱][2][ENTER].
Die oben definierte Gamma-Funktion hat die Eigenschaft, dass

$$\Gamma(\alpha) = (\alpha-1)\ \Gamma(\alpha-1), \quad \text{für } \alpha > 1.$$

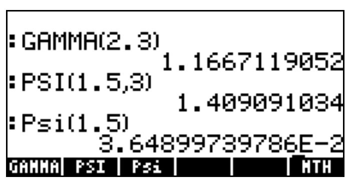
Deshalb kann diese mit der Faktoriellen einer Zahl verglichen werden, d.h.
$\Gamma(\alpha) = (\alpha-1)!$,
wenn α eine positive Integerzahl ist. Wir können die Funktion Faktorielle auch zur Berechnung der Gamma-Funktion und umgekehrt, verwenden. So z.B. $\Gamma(5) = 4!$ oder, 4 ALPHA ↱ 2 ENTER. Die Funktion Faktorielle steht im MTH-Menü, über das Menü *7. PROBABILITY..* (Wahrscheinlichkeit) zur Verfügung.

Die Funktion PSI, $\Psi(x,y)$, stellt die y-te Ableitung der Gamma-Funktion dar, d.h. $\Psi(n,x) = \frac{d^n}{dx^n}\psi(x)$, wobei $\psi(x)$ als die Digamma-Funktion oder Psi Funktion bekannt ist. Für diese Funktion muss *y* eine positive Integerzahl sein.

Die Funktion Psi, $\psi(x)$ oder Digamma-Funktion, wird als $\psi(x) = \ln[\Gamma(x)]$ definiert.

Beispiele dieser Sonderfunktionen werden sowohl im ALG- wie auch im PRN-Modus gezeigt. Als Beispiel überprüfen Sie, dass *GAMMA(2,3) = 1,166711…*, *PSI(1,5,3) = 1,40909.*, und *Psi(1,5) = 3,64899739..E-2* ist.

Diese Berechnungen werden in den nachfolgenden Screenshots dargestellt:



Konstanten des Rechners

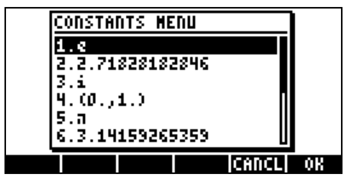
Nachfolgend die von Ihrem Rechner verwendeten mathematischen Konstanten:

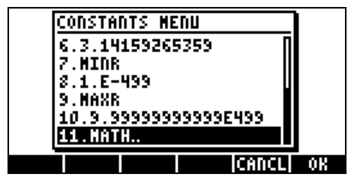
- *e*: die Basis eines natürlichen Logarithmus
- *i*: die imaginäre Einheit, $i^2 = -1$.
- π: das Verhältnis der Länge eines Kreises zu seinem Durchmesser.
- MINR: die im Rechner zur Verfügung stehende kleinste reelle Zahl

- MAXR: die im Rechner zur Verfügung stehende größte reelle Zahl

Um Zugang zu diesen Konstanten zu bekommen, wählen Sie Option *11. CONSTANTS..* im Menü MTH

Die Konstanten werden wie folgt aufgelistet:





Durch Auswahl einer dieser Einträge wird der ausgewählte Wert, entweder ein Symbol (z.B. *e*, *i*, π, *MINR*, oder *MAXR)* oder ein Wert (*2,71.., (0,1), 3,14.., 1E-499, 9,99..E499*) in den Stack ausgegeben.

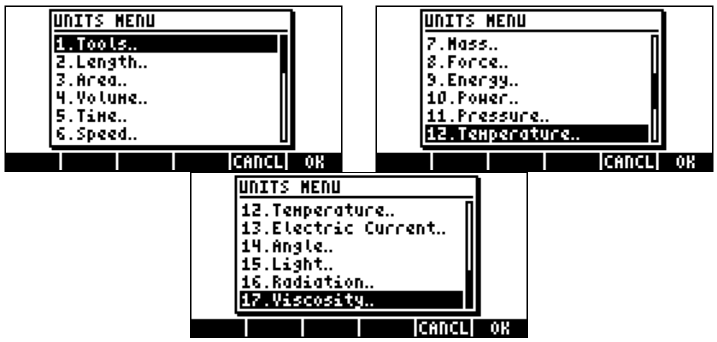
Beachten Sie, dass *e* von der Tastatur aus, als *exp(1)* zur Verfügung steht, d.h. ← e^x / ENTER, im ALG-Modus oder / ENTER ← e^x , im RPN-Modus. Auch π ist direkt von der Tastatur aus verfügbar als ← π . Schließlich ist auch *i* über die Tastatur verfügbar, und zwar über ← *i* .

Operationen mit Einheiten

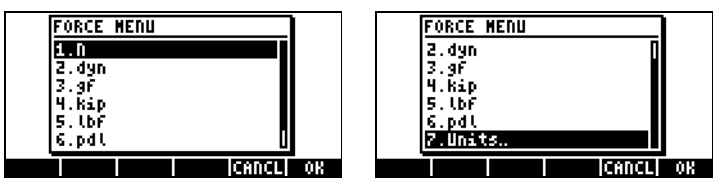
Zahlen im Rechner können unterschiedliche Einheiten zugeordnet sein. So können Sie Ergebnisse über ein konsistentes Einheiten-System berechnen und die Ergebnisse mit der geeigneten Kombination von Einheiten ausgeben lassen.

Das Menü UNITS

Das Menü UNITS wird über die Tastenkombination ↱ UNITS (der Taste 6 zugeordnet) gestartet. Mit dem System Flag 117 auf *CHOOSE* Kästchen gesetzt, erscheint das nachfolgende Menü:



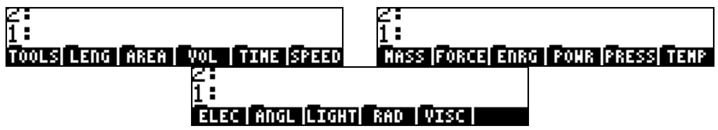
Option *1. Tools..* enthält Funktionen, welche sich auf Einheiten beziehen (werden später diskutiert). Options *3. Length..* bis *17.Viscosity..* enthalten Menüs mit einer Reihe von Einheiten für jede der beschriebenen Mengen. So z.B., wenn Sie das Menü *8. Force..* auswählen, bekommen Sie das folgende Menü von Einheiten:



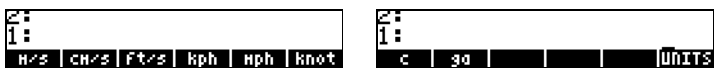
So wird der Benutzer die meisten der Einheiten (einige davon wie z.B. dyne, wird heutzutage recht selten verwendet) aus dem Physikunterricht wieder erkennen: *N* = Newton, *dyn* = Dyn, *gf* = Gramm – Kraft (um einen Unterschied für Gramm-Masse oder nur einfach Gramm als Gewichtseinheit zu machen), *kip* = Kilopond (1000 Pfund), *lbf* = Pound-Force (um vom Pfund als Gewichtseinheit zu unterscheiden), *pdl* = Poundal.

Um eine Einheit einer Zahl zuzuordnen, muss dieser Zahl ein Unterstrich folgen. So wird die Kraft von 5 N als 5_N eingegeben.

Für umfassende Berechnungen mit Einheiten, bieten die SOFT Menüs einen bequemeren Weg Einheiten zuzuordnen. Ändern Sie das System Flag 117 auf SOFT Menüs (siehe Kapitel 1) und verwenden Sie die Tastenkombination ↱ UNITS , um in folgende Menüs zu gelangen. Drücken Sie die Taste NXT, um auf die nächste Seite des Menüs zu gelangen.

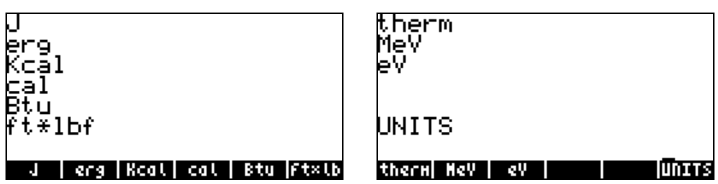


Wenn Sie die entsprechende Funktionstaste drücken, wird ein Untermenü von Einheiten zu dieser Auswahl angezeigt. So z.B. stehen für das Untermenü SPEED folgende Einheiten zur Verfügung:



Ein erneutes drücken der Funktionstaste UNITS , bringt Sie zum UNITS Menü zurück.

Beachten Sie, dass Sie jederzeit die komplette Liste der Menüeinträge durch drücken der Tastenfolge ↱▽, anzeigen können z.B. werden für das ENRG Set von Einheiten nachfolgende Einträge angezeigt:



Anmerkung: Verwenden Sie die Taste NXT oder die Tastenkombination ↰ PREV zur Navigation zwischen den einzelnen Menüs.

Zur Verfügung stehende Einheiten

Nachfolgend ist eine Liste von Einheiten, welche über das Menü UNITS zur Verfügung stehen. Erst wird das Symbol der Einheit gefolgt vom Namen der Einheit in Klammer angezeigt:

LENGTH (LÄNGE)

m (Meter), cm (Zentimeter), mm (Millimeter), yd (Yard), ft (Fuß), in (Zoll), Mpc (Mega Parsec), pc (Parsec), lyr (Lichtjahr), au (astronomische Einheit), km (Kilometer), mi (internationale Meile), nmi (Seemeile), miUS (US gesetzliche englische Meile), chain (Kette), rd (Rute), fath (Kubikfuß), ftUS (Vermessungsfuß), Mil (Mil), μ (Mikron), Å (Angström), fermi (Fermi)

AREA (FÄCHE)

m^2 (Quadratmeter), cm^2 (Quadratzentimeter), b (Barn – Maßeinheit des Wirkungsquerschnittes), yd^2 (Quadratyard), ft^2 (Quadratfuß), in^2 (Quadratzoll), km^2 (Quadratkilometer), ha (Hektar), a (Are), mi^2 (Quadratmeile), miUS^2 (gesetzliche englische Quadratmeile), acre (Acre)

VOLUME (VOLUMEN)

m^3 (Kubikmeter), st (Ster), cm^3 (Kubikzentimeter), yd^3 (Kubikyard), ft^3 (Kubikfuß), in^3 (Kubikzoll), l (Liter), galUK (UK Gallone), galC (Kanadische Gallone), gal (US Gallone), qt (Quart), pt (Pint), ml (Milliliter), cu (US Tasse), ozfl (US fluid ounce), ozUK (UK fluid ounce), tbsp (Löffel), tsp (Teelöffel), bbl (Barrel), bu (Bushel - Trockenhohlmaß), pk (Peck - Trockenmaß), fbm (Board Foot - Holzmaß)

TIME (ZEIT)

yr (Jahr), d (Tag), h (Stunde), min (Minute), s (Sekunde), Hz (Hertz)

SPEED (GESCHWINDIGKEIT)

m/s (Meter pro Sekunde), cm/s (Zentimeter pro Sekunde), ft/s (Fuß pro Sekunde), kph (Kilometer pro Stunde), mph (Meilen pro Stunde), knot (Knoten – nautische Meilen pro Stunde), c (Lichtgeschwindigkeit), ga (Gravitationsbeschleunigung)

MASS (MASSE)
kg (Kilogramm), g (Gramm), Lb (avoirdupois Pound - Handelspfund), oz (Unze), slug (Slug, Gee-pound), lbt (Troy pound), ton (short ton), tonUK (long ton), t (metric ton), ozt (Troy Unze), ct (Karat), grain (Grain), u (atomare Masseeinheit), mol (Mol)

FORCE (KRAFT)
N (Newton), dyn (Dyn), gf (Gramm-Kraft), kip (Kilopond-Kraft), lbf (pound-force), pdl (Poundal)

ENERGY (ENERGIE)
J (Joule), erg (Erg), Kcal (Kilokalorie), Cal (Kalorie), Btu (englische Kalorie - Wärmemenge), ft×lbf (Foot-Pound), therm (EEC (GB) Wärmeeinheit zur Lieferung von Stadtgas), MeV (Megaelektronen Volt), eV (Elektronenvolt)

POWER (KRAFT)
W (Watt), hp (Pferdestärke),

PRESSURE (DRUCK)
Pa (Pascal), atm (Atmosphäre), bar (bar), psi (Pfund pro Quadratzoll), torr (Torr), mmHg (Millimeter Quecksilbersäule), inHg (Zoll Quecksilbersäule), inH20 (Zoll Wassersäule),

TEMPERATUR
°C (Grad Celsius), °F (Grad Fahrenheit), K (Kelvin), °R (Grad Rankine),

ELEKTRISCHER STROM (Elektrische Maßeinheiten)
V (Volt), A (Ampere), C (Coulomb), Ω (Ohm), F (Farad), W (Watt), Fdy (Faraday), H (Henry), mho (mho), S (Siemens), T (Tesla), Wb (Weber)

ANGLE (Flächenwinkel und Raumwinkel)
° (sexagesimal Grad), r (Radian), Grad (Zentesimalgrad), arcmin (Bogenminute), arcs (Bogensekunde), sr (Steradiant)

LICHT (Maßeinheiten für das Licht)

fc (Footcandle), flam (Foot-Lambert), lx (Lux), ph (Phot), sb (Stilb), lm (Lumem), cd (Candela), lam (Lambert)

RADIATION (STRAHLUNG)

Gy (Gray), rad (Rad), rem (Rem), Sv (Sievert), Bq (Becquerel), Ci (Curie), R (Röntgen)

VISCOSITY (VISKOSITÄT)

P (Poise), St (Stokes)

Nicht aufgelistete Einheiten

Nicht aufgelistete Einheiten im Menü Units, die trotzdem im Rechner vorhanden sind: gmol (Gramm-Mol), lbmol (Pound-Mol), rpm (Umdrehungen pro Minute), dB (Dezibel). Diese Maßeinheiten erreicht man über das Menü 117.02, welches im ALG-Modus über MENU (117.02) oder im RPN-Modus unter MENU 117.02 [ENTER] gestartet wird. In der Anzeige des Menüs erhalten Sie nachfolgende Einträge (verwenden Sie dazu die Tastenfolge [→][▼], um die Beschriftung im Display anzuzeigen):

Auf diese Einheiten kann aber auch über den Katalog zugegriffen werden, so z.B.:

gmol: [→] CAT [ALPHA] [←] [G]
lbmol: [→] CAT [ALPHA] [←] [L]
rpm: [→] CAT [ALPHA] [←] [R]
dB: [→] CAT [ALPHA] [←] [D]

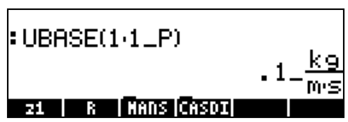
Umrechnung in Grundeinheiten

Verwenden Sie die Funktion UBASE, um jede dieser Einheiten in die Standardeinheiten des SI-Systems zu konvertieren. Um z.B. herauszufinden, welchen Wert 1 Poise (Viskositätseinheit) in SI-Einheiten darstellt, gehen Sie wie folgt vor:

Im ALG-Modus, System Flag auf 117 auf *CHOOSE* Kästchen gesetzt:

→ UNITS	Wählen Sie das Menü UNITS
OK	Wählen Sie das Menü TOOLS
▼ OK	Wählen Sie die Funktion UBASE
1 → _	Tragen Sie 1 und Unterstrich ein
→ UNITS	Wählen Sie das Menü UNITS
▲ OK	Wählen Sie die Option VISCOSITY
OK	Wählen Sie das Menü UNITS
ENTER	Konvertieren Sie die Einheiten

Als Ergebnis erhalten Sie die folgende Anzeige (d.h. *1 Poise = 0,1 kg/(m·s)*):



Im RPN-Modus, System Flag auf 117 auf *CHOOSE* Kästchen gesetzt:

1	Tragen Sie 1 (kein Unterstrich) ein
→ UNITS	Wählen Sie das Menü UNITS
▲ OK	Wählen Sie die Option VISCOSITY
OK	Wählen Sie die Einheit P (Poise)
→ UNITS	Wählen Sie das Menü UNITS
OK	Wählen Sie das Menü TOOLS
▼ OK	Wählen Sie die Funktion UBASEIm ALG-Modus,

System Flag auf 117 auf *SOFT Menüs* gesetzt.

→ UNITS	Wählen Sie das Menü UNITS
TOOLS	Wählen Sie das Menü TOOLS
UBASE	Wählen Sie die Funktion UBASE
1 → _	Tragen Sie 1 und Unterstrich ein
→ UNITS	Wählen Sie das Menü UNITS
← PREV VISC	Wählen Sie die Option VISCOSITY
P	Wählen Sie die Einheit P (Poise)

ENTER Konvertieren Sie die Einheiten

Im RPN-Modus, System Flag auf 117 auf *SOFT Menüs* gesetzt.

1	Tragen Sie 1 (kein Unterstrich) ein
↱ UNITS	Wählen Sie das Menü UNITS
↰ PREV VISC	Wählen Sie die Option VISCOSITY
P	Wählen Sie die Einheit P (Poise)
↱ UNITS	Wählen Sie das Menü UNITS
TOOLS	Wählen Sie das Menü TOOLS
UBASE	Wählen Sie die Funktion UBASE

Einheiten den Zahlen zuordnen

Um eine Einheit einer Zahl zuzuordnenUm eine Einheit einer Zahl zuzuordnen, muss ein Unterstrich an diese Zahl angehängt werden (↱ _ , Taste(8,5)). So wird die Kraft von 5 N als 5_N eingegeben.
Nachfolgend die Tastenfolge, die im ALG-Modus, mit System Flag 117 auf *CHOOSE* Kästchen gesetzt, eingegeben werden muss:

5 ↱ _	Tragen Sie die Zahl und den Unterstrich ein
↱ UNITS	Begeben Sie sich in das Menü UNITS
8 OK	Wählen Sie die Krafteinheiten (*8. Force..*)
OK	Wählen Sie Newton (*N*)
ENTER	Geben Sie die Anzahl der Einheiten in den Stack ein

Die Anzeige sieht wie folgt aus:

Anmerkung: Vergessen Sie den Unterstrich, wird das Ergebnis 5*N aussehen, wobei N eine mögliche Variable darstellt, nicht aber die Einheit Newton.

Verwenden Sie nachfolgende Tastenfolge, um das Gleiche im RPN-Modus einzugeben:

5	Geben Sie die Zahl ein (geben Sie keinen

Unterstrich ein)

[→] UNITS	Begeben Sie sich in das Menü UNITS
[8] OK	Wählen Sie die Krafteinheiten (*8. Force..*)
OK	Wählen Sie Newton (*N*)

Beachten Sie dabei, dass der Unterstrich automatisch eingefügt wird, wenn der RPN-Modus aktiviert ist. Das Ergebnis ist die nachfolgende Anzeige:

Wie vorhin angedeutet, wenn das System Flag 117 auf SOFT Menüs steht, wird das Menü UNITS als Bezeichnung für die Funktionstasten angezeigt. Diese Einstellung ist für umfassende Berechnungen sehr praktisch.

Nachfolgend die Tastenfolge zur Eingabe von Einheiten mit ausgewählter Option SOFT-Menü, im ALG- und im PRN-Modus: Um im ALG-Modus den Ausdruck 5_N einzugeben, verwenden Sie die Tastenfolge:

[5] [→] _	Tragen Sie die Zahl und den Unterstrich ein
[→] UNITS	Begeben Sie sich in das Menü UNITS
[NXT] FORCE	Wählen Sie die Einheiten für die Kraft
N	Wählen Sie Newton (*N*)
[ENTER]	Geben Sie die Anzahl der Einheiten in den Stack ein

Um den gleichen Ausdruck im RPN-Modus einzugeben, verwenden Sie die Tastenfolge:

[5]	Tragen Sie die Zahl (ohne Unterstrich) ein
[→] UNITS	Begeben Sie sich in das Menü UNITS
[NXT] FORCE	Wählen Sie die Einheiten für die Kraft
N	Wählen Sie Newton (*N*)

Anmerkung: Sie können einen Ausdruck mit Einheiten eingeben, indem Sie den Unterstrich und die Einheiten über die [ALPHA]–Taste eingeben, [5][→] _ [ALPHA][N] ergibt z.B. den folgenden Eintrag: 5_N

Vorzeichen von Einheiten

Für Einheiten können Sie Vorzeichen gemäß der nachfolgenden Tabelle aus dem SI System eingeben.
Als erstes wird die Abkürzung des Vorzeichens aufgeführt, anschließend der Name, gefolgt vom Exponenten x im Faktor 10^x, welcher dem jeweiligen Vorzeichen entspricht:

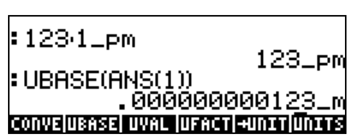
Vorzeichen	Name	x	Vorzeichen	Name	x
Y	Yotta	+24	d	Deci	-1
Z	Zetta	+21	c	Centi	-2
E	exa	+18	m	Milli	-3
P	Peta	+15	µ	Micro	-6
T	Terra	+12	n	Nano	-9
G	Giga	+9	p	Pico	-12
M	Mega	+6	f	Femto	-15
k,K	Kilo	+3	a	Atto	-18
h,H	Hekto	+2	z	Zepto	-21
D(*)	Deka	+1	y	Yocto	-24

(*) Im SI System, wird höchstwahrscheinlich dieses Vorzeichen (*da*) anstelle von *D* stehen. Benutzen Sie dennoch das D für deka im Rechner.

Um diese Vorzeichen einzugeben, tippen Sie einfach das Vorzeichen über die ALPHA Tastatur ein. Um z.B. 123 pm (1 Picometer) einzugeben, tippen Sie wie folgt:

1 2 3 ↱ _ ALPHA ↰ P ALPHA ↰ M

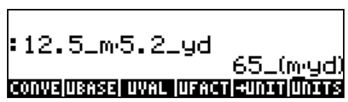
Benutzen Sie UBASE, um in die Standard-Einheit (1 m) umzuwandeln; das Ergebnis sieht so aus:



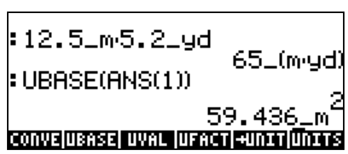
Operationen mit Einheiten

Sobald eine Menge gefolgt von Einheiten in den Stack eingegeben wurde, kann diese in Berechnungen, ähnlich reiner Zahlen, verwendet werden, ausgenommen, dass Mengen mit Einheiten nicht als Argumente in Funktionen eingesetzt werden können (sagen wir SQ oder SIN). Versuchen wir LN(10_m) zu berechnen, bekommen wir eine Fehlermeldung: *Error: Bad Argument Type. (Fehler: ungültiger Argumenttyp)*

Hier sind einige Berechnungsbeispiele im ALG-Modus. Seien Sie bei der Multiplikation und Division von Mengen mit Einheiten vorsichtig, Sie müssen jede Menge mit der dazugehörigen Einheit in Klammer setzen. Um z.B. das Produkt 12,5m × 5,2 yd einzugeben, muss Ihre Eingabe so aussehen (12,5_m)*(5,2_yd) ENTER :

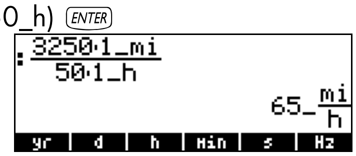


welche dann als 65_(m·yd) angezeigt wird. Benutzen Sie die Funktion UBASE, um in die Einheiten des SI-Systems zu konvertieren:

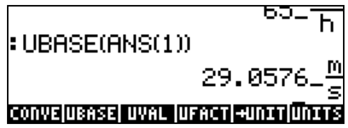


Anmerkung: Merken Sie sich, dass die Variable ANS(1) über die Tastenkombination ← ANS (der Taste ENTER zugeordnet) aufgerufen werden kann.

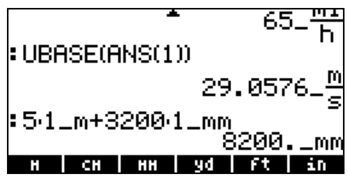
Um eine Division durchzuführen, sagen wir 3250 mi / 50 h, geben Sie diese wie folgt ein: (3250_mi)/(50_h) ENTER



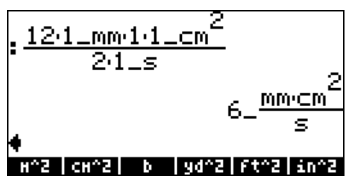
welche dann, mit der UBASE Funktion in SI-Einheiten konvertiert folgendes ergibt:



Additionen und Subtraktionen können im ALG-Modus, ohne Eingabe von Klammern durchgeführt werden, z.B. 5 m + 3200 mm, können ganz einfach als 5_m + 3200_mm ENTER eingegeben werden.



Kompliziertere Ausdrücke hingegen benötigen Klammern, so z.B. (12_mm)*(1_cm^2)/(2_s) ENTER:



Bei Stack Berechnungen im PRN-Modus, werden keine Klammern bei der Eingabe verschiedener Ausdrücke benötigt, so z.B.

(12_m) ENTER (1,5_yd) ENTER ×
(3250_mi) ENTER (50_h) ENTER ÷

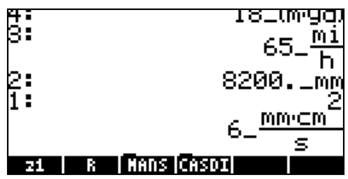
Diese Operationen ergeben folgende Ausgabe:

Versuchen Sie auch nachfolgende Operationen:

5_m ENTER 3200_mm ENTER +

12_mm ENTER 1_cm^2 ENTER × 2_s ENTER ÷

Diese letzten beiden Operationen ergeben folgende Ausgabe:



Anmerkung: In Ausdrücke des EquationWriters können keine Einheiten eingegeben werden.

Werkzeuge zur Manipulation von Einheiten

Das Menü UNITS enthält ein Untermenü TOOLS, welches folgende Funktionen zur Verfügung stellt:

CONVERT(x,y):	konvertiert Objekteinheit x in Einheiten des Objektes y
UBASE(x):	konvertiert Objekteinheit x in SI-Einheiten
UVAL(x):	extrahiert den Wert aus Objekteinheit x
UFACT(x,y):	Multipliziert eine Einheit x aus der Objekteinheit x
→UNIT(x,y):	kombiniert Werte von x mit Einheiten von y

Die Funktion UBASE wurde im Detail in einem früheren Abschnitt dieses Kapitels beschrieben. Um auf irgendeine dieser Funktionen zuzugreifen, gehen Sie wie in den vorangegangen gezeigten UBASE Beispielen vor. Beachten Sie, während die Funktion UVAL ein einziges Argument benötigt, werden für die Funktionen CONVERT, UFACT und →UNIT jeweils zwei Argumente benötigt.

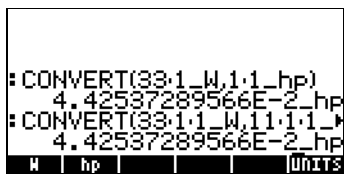
Versuchen Sie die nachfolgenden Übungen, in der von Ihnen vorgezogenen Einstellung Ihres Rechners. Die nachfolgende Ausgabe wurde im ALG-Modus, mit System Flag 117 auf *SOFT* Menü eingestellt, erzeugt:

CONVERT Beispiele

Beide Beispiele ergeben das gleiche Ergebenis, d.h. um 33 Watt in btu's zu konvertieren

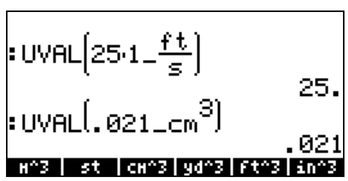
CONVERT(33_W,1_hp) ENTER
CONVERT(33_W,1_hp) ENTER

Diese Operationen werden in der Anzeige wie nachfolgend angezeigt:



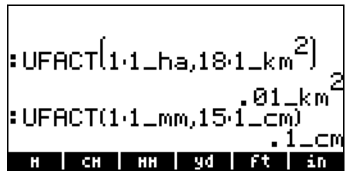
UVAL Beispiele:

UVAL(25_ft/s) ENTER
UVAL(0.021_cm^3) ENTER



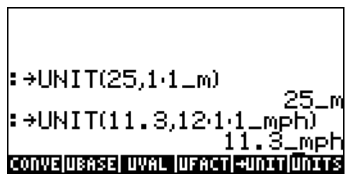
UFACT Beispiele:

UFACT(1_ha,18_km^2) ENTER
UFACT(1_mm,15,1_cm) ENTER



→UNIT Beispiele

→UNIT(25,1_m) [ENTER]
→UNIT(11,3,1_mph) [ENTER]



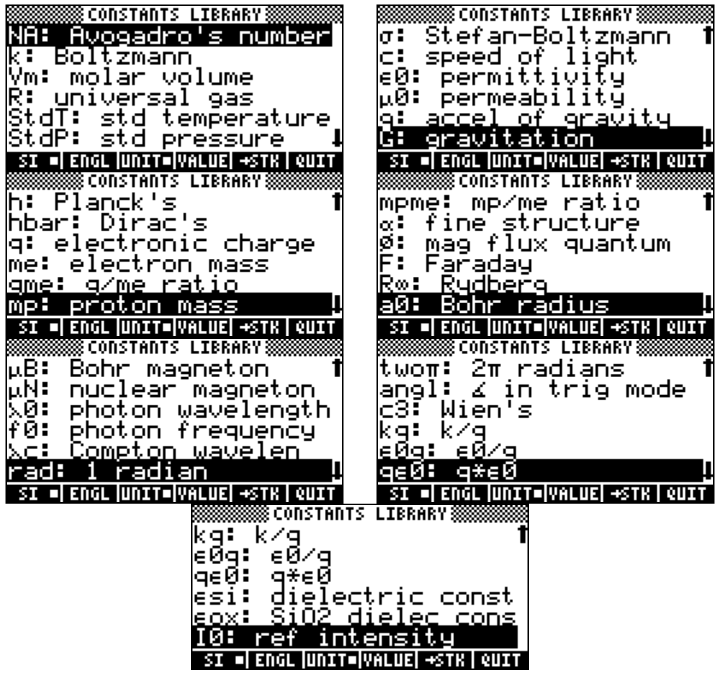
Physische Konstanten im Rechner

Genau wie die Behandlung von Einheiten, erörtern wir auch die im Rechner zur Verfügung stehenden physikalischen Konstanten. Die physikalischen Konstanten des Rechners befinden sich in einer *constants library* (Konstantenbibliothek), welche mit dem Befehl CONLIB aufgerufen werden können. Um diesen Befehl zu starten könnten Sie folgendes einfach in den Stack eingeben:

[ALPHA][ALPHA][C][O][N][L][I][B][ALPHA][ENTER]

oder Sie wählen den Befehl CONLIB aus dem Befehle Katalog, wie folgt: Starten Sie zuerst den Katalog wie folgt: [→] _CAT [ALPHA][C]. Benutzen Sie dann die Pfeiltasten [▲][▼], um CONLIB auszuwählen. Anschießend drücken Sie die Funktionstaste [F6] ([OK]) . Falls nötig, drücken Sie [ENTER].

Die Anzeige der Konstantenbibliothek sieht wie folgt aus (benutzen die Pfeiltasten zur Navigation in der Bibliothek):

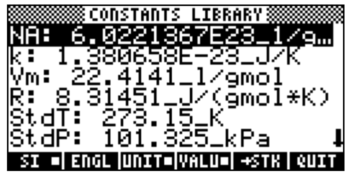


Die dieser Anzeige zugeordneten Funktionstasten der CONSTANTS LIBRARY, enthalten folgende Funktionen:

SI	wenn ausgewählt, werden die Werte der Konstanten in SI-Einheiten angezeigt
ENGL	wenn ausgewählt, werden die Werte der Konstanten in Englischen-Einheiten angezeigt (*)
UNIT	wenn ausgewählt, werden die Konstanten zusammen mit den ihnen zugeordneten Einheiten ausgegeben (*)
VALUE	wenn ausgewählt, werden die Konstanten ohne Einheiten ausgegeben
→STK	kopiert den Wert (mit oder ohne Einheiten) in den Stack
QUIT	damit verlassen Sie die Konstantenbibliothek

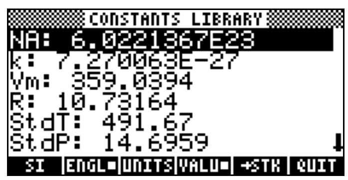
(*) Aktive nur, wenn die Funktion VALUE aktiv ist.

So sieht die erste Seite der CONSTANTS LIBRARY Anzeige aus, wenn die Option VALUE ausgewählt wurde (Einheiten im SI-System):



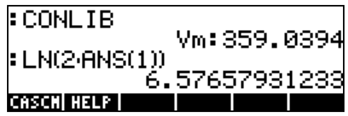
Um die Werte der Konstanten im Englischen (oder Imperial) System anzuzeigen, drücken Sie die Option ENGL :

Schalten wir die UNITS Option aus, (deselektieren diese, drücken Sie die Taste UNITS) werden nur die Werte angezeigt (in diesem Fall wurden Englische Einheiten ausgewählt):



Um den Wert Vm in den Stack zu kopieren, wählen Sie einen Variablen-Namen und drücken Sie erst die Taste !→STK dann QUIT. Ist der Rechner auf ALG-Modus eingestellt, sieht die Anzeige wie folgt aus:

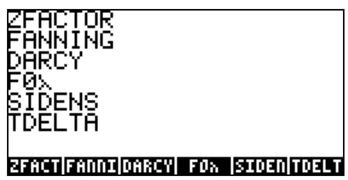
Die Anzeige zeigt einen sogenannten *tagged value (*gekennzeichneten Wert) `Vm:359,0394`. In diesem Fall ist Vm die Kennzeichnung des Ergebnisses. Jede arithmetische Operation mit dieser Nummer wird die Markierung ignorieren. Versuchen Sie z.B.: [↱] LN [2] [×] [↰] ANS [ENTER], wodurch folgendes Resultat erzeugt wird:



Bei der gleichen Berechnung im RPN-Modus benutzen Sie folgende Tastenfolgen (nachdem der Wert von Vm aus der Konstantenbibliothek extrahiert wurde): [2] [ENTER] [×] [↱] LN

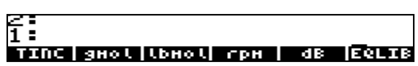
Spezielle physikalische Funktionen

Menu 117, ausgewählt über MENU (117) im ALG-Modus oder MENU 117 [ENTER] im RPN-Modus, erzeugt das nachfolgende Menü (Beschriftungen werden über die Tasten [↱] [▼] angezeigt):



Die Funktionen schließen ein:
ZFACTOR: Gaskompressibilität Z Faktor Funktion
FANNING: Widerstandsfaktor der Strömungs-Auffächerung
DARCY: Darcy-Weisbach Widerstandsfaktor der Strömungs-Auffächerung
F0λ: Ausströmende Kraftfunktion Schwarzer Körper (Planckscher Strahler)
SIDENS: innere Dichte Silizium
TDELTA: Delta Funktion Temperatur

Auf der zweiten Seite dieses Menüs (drücken Sie [NXT]) finden wir nachfolgende Elemente:



Auf dieser Menüseite ist eine Funktion (TINC) und eine Anzahl von Maßeinheiten, die in einem früheren Abschnitt beschrieben wurden (siehe oben). Die interessante Funktion ist:

TINC: Befehl Wertzuwachs (Inkrement) für Temperatur

Von all den Funktionen aus diesem Menü (UTILITY), d.h. ZFACTOR, FANNING, DARCY, F0λ, SIDENS, TDELTA und TINC, werden die Funktionen FANNING und DARCY in Kapitel 6, im Zusammenhang mit Gleichungen zur Durchflussberechnung in Rohrleitungen beschrieben. Für die noch verbleibenden Funktionen finden Sie nachfolgend eine kurze Beschreibung.

Funktion ZFACTOR

Die Funktion ZFACTOR berechnet den Berichtigungsfaktor für die Gaskompressibilität bei nicht idealeam Verhalten von Kohlewasserstoffgas. Die Funktion wird über ZFACTOR(x_T, y_P) aufgerufen, wobei x_T die verringerte Temperatur, d.h. das Verhältnis der eigentlichen Temperatur zur pseudo-kritischen Temperatur ist und y_P der verringerte Druck, d.h. das Verhältnis des eigentlichen Drucks zum pseudo-kritischen Druck darstellt. Der Wert von x_T muss zwischen 1,05 und 3,0 liegen, während der Wert von y_P zwischen 0 und 30 liegen muss. Beispiel im ALG-Modus:

Funktion F0λ

Die Funktion F0λ (T, λ) berechnet den Bruch (dimensionslos) der gesamten ausströmenden Kraft der schwarzen Körper bei einer Temperatur T zwischen den Wellenlängen 0 und λ. Sind keine Einheiten T und λ zugewiesen, wird angenommen, dass T in K und λ in m ist. Beispiel im ALG-Modus:

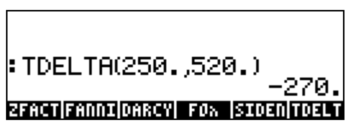
Funktion SIDENS

Die Funktion SIDENS(T) berechnet die innere Dichte von Silizium (in Einheiten von 1/cm^3) als Funktion der Temperatur T (T in K), für T zwischen 0 und 1685 K. So zum Beispiel:

Funktion TDELTA

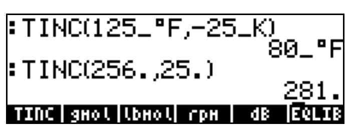
Die Funktion TDELTA(T_0,T_f) liefert den Temperaturzuwachs $T_f - T_0$. Das Ergebnis wird in der gleichen Maßeinheit, falls überhaupt, als T_0 ausgegeben. Andernfalls erhalten Sie einfach die Zahlendifferenz. So zum Beispiel:

Der Sinn dieser Funktion ist es eine Temperaturdifferenzberechnung, bei unterschiedlichen Maßeinheiten der Temperatur, zu vereinfachen. Andernfalls wird einfach nur eine Differenz der Zahlen berechnet. z.B.



Funktion TINC

Die Funktion TINC(T_0,ΔT) berechnet T_0+DT. Diese Funktion ist TDELTA insofern ähnlich, dass das Ergebnis in der Maßeinheit T_0 ausgegeben wird. Andernfalls ist das erhaltene Ergebnis eine einfache Addition dieser Werte, z.B.



Definieren und anwenden von Funktionen

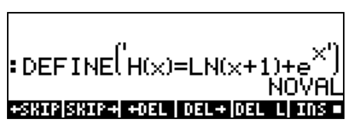
Der Benutzer kann seine eigenen Funktionen selbst über den Befehl DEF, welchen er über die Tastenfolge [←] DEF (der Taste [2] zugeordnet) erreicht, aufruft. Die Funktion muss in nachfolgendem Format eingegeben werden:

Function_name(arguments) = expression_containing_arguments

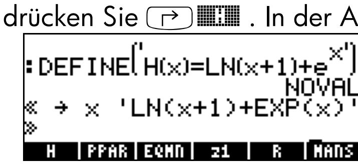
Als Beispiel könnten wir eine einfache Funktion definieren *H(x) = ln(x+1) + exp(-x)*.
Angenommen Sie müssen diese Funktion für eine Zahl von diskreten (unstetigen) Werten auswerten, deshalb möchten Sie nur eine einzige Taste dazu benutzen und das gewünschte Ergebnis, ohne umständliches separates eintippen der einzelnen Werte auf der rechten Seite. Im nachfolgenden Beispiel, wird angenommen, dass sich der Rechner im ALG-Modus befindet. Geben Sie die nachfolgende Tastenfolge ein:

[←] DEF ['] [ALPHA] [H] [←] () [ALPHA] [←] [X] [▶] [→] =
[→] LN [ALPHA] [←] [X] [+] [1] [▶] [+] [←] e^x [ALPHA] [←] [X] [ENTER]

Die Anzeige sieht wie folgt aus:



Drücken Sie die Taste [VAR]; Sie werden feststellen, dass sich eine neue Variable in Ihrer Funktionstaste ([H]) befindet. Um sich den Inhalt dieser Variablen anzuzeigen drücken Sie [→] [H] . In der Anzeige erscheint nun:



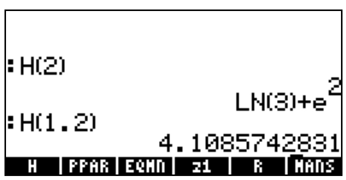
Somit enthält nun die Variable H ein Programm:

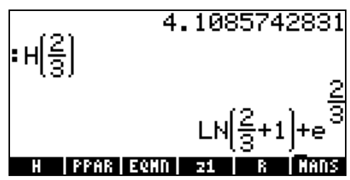
<< → x 'LN(x+1) + EXP(x)' >>

Dies ist ein einfaches Programm in der Standard-Programmiersprache des HP 48, welches auch in die neue Reihe HP 49 G integriert wurde. Diese Programmiersprache wird UserRPL bezeichnet. Das oben aufgeführte Programm ist relativ einfach und besteht aus zwei Teilen, welche sich zwischen den Programm-Containern « » befinden:

- Eingabe: → x → x
- Prozess: 'LN(x+1) + EXP(x) '

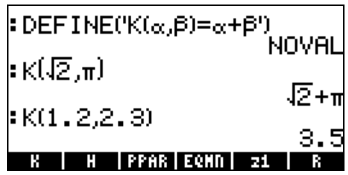
Dies wird so interpretiert: trage einen Wert ein, der temporär dem Namen x (als lokale Variable bezeichnet) zugeordnet wird, werte den Ausdruck zwischen den Anführungszeichen aus, welche lokale Variablen enthalten, zeige dann den berechneten Ausdruck.
Um die Funktion im ALG-Modus zu starten, geben Sie den Namen der Funktion, gefolgt vom Argument in Klammern ein, z.B. H ← () 2 ENTER.
Nachfolgend einige Beispiele:





Im RPN-Modus müssen Sie zuerst das Argument eingeben, dann die Funktionstaste, welche der Variablen mit dem Namen H entspricht, drücken, erst dann wird die Funktion gestartet. Sie könnten z.B. versuchen: 2 ENTER H . Die anderen oben aufgeführten Beispiele können wie folgt eingegeben werden: 1 . 2 ENTER H , 2 ÷ 3 ENTER H .

Funktionen können aber auch mehr als zwei Argumente haben. So z.B. zeigt die untere Abbildung die Definition der Funktion $K(\alpha,\beta) = \alpha+\beta$ und ihrer Auswertung mit den Argumenten $K(\sqrt{2},\pi)$ und $K(1.2,2.3)$:



Die Inhalte der Variablen K sind: << → α β 'α+β' >>.

Funktionen die über mehr als einen Ausdruck definiert werden

In diesem Abschnitt behandeln wir Funktionen die von zwei oder mehreren Ausdrücken definiert werden. Ein Beispiel solcher Funktionen wäre:

$$f(x) = \begin{Bmatrix} 2 \cdot x - 1, & x < 0 \\ x^2 - 1, & x > 0 \end{Bmatrix}$$

Im HP 49 G steht die Funktion IFTE (IF-Then-Else) zur Verfügung, welche derartige Funktionen beschreiben würde.

Die Funktion IFTE

Die IFTE Funktion wird als IFTE (*condition*, *operation_if_true*, *operation_if_false - Bedingung, Aktion-wenn-wahr, Aktion-wenn-falsch*) geschrieben
Wenn die Bedingung wahr ist wird die *Bedingung-wenn-wahr* ausgeführt, ansonsten die *Bedigung-wenn-falsch.* So z.B. können wir, um die obige Funktion zu beschreiben 'f(x) = IFTE(x>0, x^2-1, 2*x-1)' schreiben. Die Funktion IFTE befindet sich im Befehle Katalog (↱ CAT). Das Symbol '>' (größer als) is vorhanden als (der Taste 1/x zugeordnet). Verwenden Sie nachfolgende Befehlsfolge, um diese Funktion im ALG-Modus zu definieren: DEF(f(x) = IFTE(x>0, x^2-1, 2*x-1))
anschließend drücken Sie ENTER. Im RPN-Modus geben Sie Definition der Funktion zwischen Apostrophen ein:

'f(x) = IFTE(x>0, x^2-1, 2*x-1)'

drücken Sie dann ↰ DEF .

Drücken Sie VAR, um ins Variablen Menü zurückzukehren. In Ihrem Funktionstastenmenü sollte die Variable f zur Verfügung stehen. Drücken Sie nun ↱ f , um das resultierende Programm zu sehen:

<< → x 'IFTE(x>0, x^2-1, 2*x-1)' >>

Um diese Funktion im ALG-Modus zu berechnen, geben Sie den Funktionsnamen, f gefolgt von der Zahl in der Sie den Ausdruck auswerten möchten ein, z.B. f(2), drücken Sie dann ENTER. Im RPN-Modus geben Sie eine Zahl ein und drücken dann f. Überprüfen Sie z.B., dass *f(2) = 3*, während *f(-2) = -5*.

Kombinierte IFTE Funktionen

Um eine kompliziertere Funktion wie die nachfolgende zu programmieren

$$g(x) = \begin{cases} -x, & x < -2 \\ x+1, & -2 \le x < 0 \\ x-1, & 0 \le x < 2 \\ x^2, & x \ge 2 \end{cases}$$

können Sie unterschiedliche Stufen der Funktion IFTE kombinieren, d.h.

'g(x) = IFTE(x<-2, -x, IFTE(x<0, x+1, IFTE(x<2, x-1, x^2)))',

Definieren Sie die Funktion über eine der oben vorgestellten Möglichkeiten und überprüfen Sie, ob g(-3) = 3, g(-1) = 0, g(1) = 0, g(3) = 9 ergibt.

Kapitel 4
Berechnungen mit komplexen Zahlen

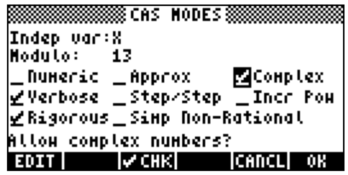
In diesem Kapitel finden Sie Beispiele von Berechnungen und Anwendungen von Funktionen mit komplexen Zahlen.

Definitionen

Eine *komplexe Zahl z* ist eine als $z = x + iy$ geschriebene Zahl, wobei x und y reelle Zahlen sind und i die *imaginäre Einheit*, definiert durch $i^2 = -1$ darstellt. Die Zahl $x+iy$ hat einen *reellen Teil* $x = Re(z)$ und einen *imaginären Teil* $y = Im(z)$. Wir können uns eine komplexe Zahl als ein Punkt $P(x,y)$ in der x-y Ebene vorstellen, wobei die x-Achse als *reelle Achse* und die y-Achse als die *imaginäre Achse* bezeichnet wird. Somit wird eine komplexe Zahl, in Form von $x+iy$, als deren *Kartesische Darstellung* bezeichnet. Eine alternative Kartesische Darstellung ist das geordnete Paar $z = (x,y)$. Die *polare Darstellung* einer komplexen Zahl lautet $z = re^{i\theta} = r\cdot\cos\theta + i\, r\cdot\sin\theta$, wobei $r = |z| = \sqrt{x^2 + y^2}$ die *Magnitude* der komplexen Zahl ist und z, and $\theta = Arg(z) = \arctan(y/x)$ das Argument der komplexen Zahl z darstellt. Das Verhältnis zwischen der Kartesischen und der polaren Darstellung einer komplexen Zahl ergibt sich aus der *Euler Formel:* $e^{i\theta} = \cos\theta + i \sin\theta$. Die *konjugiert komplexe Zahl* einer komplexen Zahl $z = x + iy = re^{i\theta}$, is $\bar{z} = x - iy = re^{-i\theta}$. Die konjugiert komplexe Zahl von i kann als Spiegelbild von z auf der reellen (x) Achse betrachtet werden. Ähnlich kann die Negative von z, $-z = -x-iy = -re^{i\theta}$, als die Spiegelung von z zum Ursprung betrachtet werden.

Umstellen des Rechners auf den COMPLEX Modus

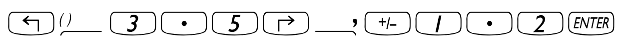
Bei der Arbeit mit komplexen Zahlen ist es von Vorteil auch den Rechner in dem Komplex-Modus umzustellen, verwenden Sie dazu die Tastenfolge:
MODE CAS 2 ▼ ▼ ▶ CHK
Der COMPLEX Modus wird in der CAS MODES Anzeige ausgewählt, indem die Option _Complex mit einem Häkchen versehen wird, d.h.



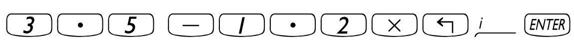
Drücken Sie OK zweimal, um zum Stack zurückzukehren.

Eingabe von komplexen Zahlen

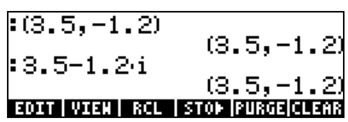
Komplexe Zahlen können in eine der beiden Kartesischen Darstellungsweisen in den Rechner eingegeben werden, und zwar entweder *x+iy* oder *(x,y)*. Die Ergebnisse im Rechner werden im Format als geordnete Paare, d.h. als *(x,y)* dargestellt angezeigt. Im ALG-Modus z.B. wird die komplexe Zahl *(3,5,-1.2)*, wie folgt eingegeben:



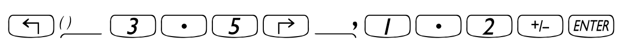
Eine komplexe Zahl kann aber auch als *x+iy* eingegeben werden. Im ALG-Modus wird *3,5-1,2i* wie folgt eingegeben:



Nach Eingabe der komplexen Zahlen erhalten Sie folgende Anzeige:



Im RPN-Modus können diese Zahlen über folgende Tastenfolge eingegeben werden:



(Beachten Sie, dass die Taste Vorzeichen ändern nach der Zahl 1,2 eingegeben wird, und zwar genau gegenteilig zum ALG-Modus Beispiel) und

(Beachten Sie auch, dass Sie im RPN-Modus einen Apostroph vor der Zahl 3,5-1,2i eingeben müssen.) Im RPN-Modus sieht die Anzeige wie folgt aus:

Beachten Sie, dass die letzte Eingabe eine komplexe Zahl im Format *x+iy* ist. Dies ist deshalb so, weil die Zahl zwischen Apostrophe eingegeben wurde, und somit einen algebraischen Ausdruck darstellt. Verwenden Sie die Taste EVAL (EVAL), um diese Zahl zu berechnen.

Sobald der algebraische Ausdruck berechnet wurde, stellen Sie die komplexe Zahl wieder her (3,5,1,2).

Polare Darstellung von komplexen Zahlen

Das obige Ergebnis zeigt eine Kartesische (rechtwinklige) Darstellung der komplexen Zahl 3,5-1,2i.   Eine polare Darstellung ist möglich, wenn wir das Koordinatensystem auf zylindrisch oder polar über die Funktion CYLIN ändern. Sie finden diese Funktion im Katalog (⮣ _CAT_).   Ändern wir auf polare Darstellung, erhalten wir folgendes Ergebnis:

Für dieses Ergebnis wurde das rechtwinklige Maß auf Radian umgestellt (Mit Hilfe der Funktion RAD können Sie jederzeit auf Radian umstellen). Das obige Ergebnis stellt eine Magnitude von 3,7 und einen Winkel von 0,33029… dar. Das Winkelsymbol (∠) wird vor dem Winkelmaß angezeigt.

Zur Kartesischen Darstellung oder rechtwinkligen Koordinaten kommen Sie mit der Funktion RECT wieder zurück (zu finden im Katalog ⮣ _CAT_).   Eine komplexe Zahl in polarer Darstellung wird als $z = r \cdot e^{i\theta}$ eingegeben.   Die Eingabe dieser komplexen Zahl kann in den Rechner über ein geordnetes

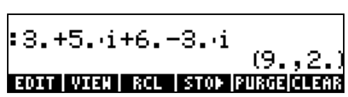
Paar erfolgen, das wie folgt aussieht (r, ∠θ). Das Winkelsymbol (∠) kann als ALPHA ↱ 6 eingegeben werden. So z.B. kann die komplexe Zahl z = $5,2e^{1.5i}$ wie folgt eingegeben werden (die Zahlen stellen den RPN Stack, vor und nach Eingabe der Zahl, dar):

Da das Koordinatensystem auf rechtwinklige (oder Kartesische) Darstellung eingestellt ist, konvertiert der Rechner die eingegebene Zahl in Kartesische Koordinaten, d.h. x = r cos θ, y = r sin θ, in diesem Fall (0,3678…, 5,18…).

Andererseits, ist das Koordinatensystem auf zylindrisch (über die Funktion CYLIN) eingestellt, bekommen Sie eine polare Darstellung bei der Eingabe einer komplexen Zahl (x,y), wobei x und y reelle Zahlen sind. Geben Sie z.B. in zylindrische Koordinaten die Zahl (3.,2.) ein. Nachfolgend können Sie den RPN Stack, vor und nach Eingabe dieser Zahl sehen:

Einfache Operationen mit komplexen Zahlen

Komplexe Zahlen können über die vier Grundrechenarten (+ − × ÷) kombiniert werden. Die Ergebnisse befolgen die algebraischen Regeln nach der Formel, dass $i^2 = -1$ ist. Operationen mit komplexen Zahlen ähneln denen mit reellen Zahlen. Versuchen Sie mit dem Rechner im ALG-Modus und das CAS auf *Complex* eingestellt, folgende Zahlen einzugeben: (3+5i) + (6-3i):



Beachten Sie, dass die reellen Teile (3+6) mit den imaginären Teilen (5-3) kombiniert werden und das erhaltene Ergebnis ein geordnetes Zahlenpaar mit dem reellen Teil 9 und dem imaginären Teil 2 darstellt. Versuchen Sie nachfolgende Berechnungen selbst:

(5-2i) - (3+4i) = (2,-6)
(3-i)·(2-4i) = (2,-14)
(5-2i)/(3+4i) = (0.28,-1.04)
1/(3+4i) = (0.12, -0.16)

Anmerkungen:
Das Produkt zweier Zahlen wird wie nachfolgend dargestellt: $(x_1+iy_1)(x_2+iy_2) = (x_1x_2 - y_1y_2) + i\,(x_1y_2 + x_2y_1)$.
Die Division zweier komplexer Zahlen wird erreicht, wenn man sowohl den Zähler als auch den Nenner mit der konjugiert komplexen Zahl des Nenners multipliziert, d.h.

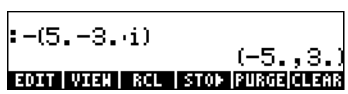
$$\frac{x_1+iy_1}{x_2+iy_2} = \frac{x_1+iy_1}{x_2+iy_2}\cdot\frac{x_2-iy_2}{x_2-iy_2} = \frac{x_1x_2+y_1y_2}{x_2^2+y_2^2} + i\cdot\frac{x_2y_1-x_1y_2}{x_2^2+y_2^2}$$

Somit ist die Inverse INV (wird mit der Taste 1/x aktiviert) definiert als:

$$\frac{1}{x+iy} = \frac{1}{x+iy}\cdot\frac{x-iy}{x-iy} = \frac{x}{x^2+y^2} + i\cdot\frac{y}{x^2+y^2}$$

Änderung des Vorzeichens einer komplexen Zahl

Das Vorzeichen einer komplexen Zahl kann mit der Taste +/- geändert werden, z.B. -(5-3i) = -5 + 3i



Eingabe der Einheit imaginäre Zahl

Um die Zahl des Typs imaginäre Einheit einzugeben, verwenden Sie ← i

Beachten Sie dabei, dass die Zahl *i* als geordnetes Zahlenpaar (0,1) eingegeben wird, wenn das CAS im APPROX Modus steht. Im EXACT Modus wir die Zahl des Typs imaginäre Einheit als *i* eingegeben.

Weitere Operationen

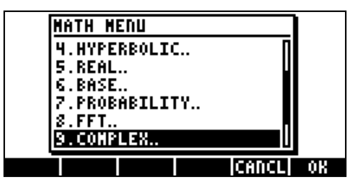
Operationen wie Magnitude, Argumente, reelle und imaginäre Teile aber auch konjugiert komplexe Zahlen werden weiter unten innerhalb des Menüs CMPLX im Detail erörtert.

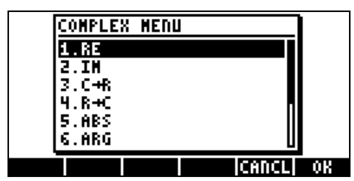
Die CMPLX Menüs

Im Rechner stehen zwei CMPLX (CoMPLeX - komplex) Menüs zur Verfügung. Eines steht über das Menü MTH (in Kapitel 3 vorgestellt) und eines direkt über die Tastatur (→ CMPLX) zur Verfügung. Nachfolgend werden beide CMPLX Menüs vorgestellt.

CMPLX Menü über das Menü MTH

Angenommen das System Flag 117 ist auf **CHOOSE Kästchen** (siehe Kapitel 2) eingestellt, wird das CMPLX-Untermenü innerhalb des MTH-Menüs wie folgt aufgerufen: ← MTH 9 OK . Die nachfolgende Sequenz von Screenshots veranschaulichen diese Schritte:



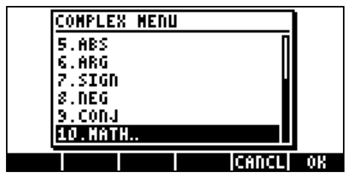


Das erste Menü (Optionen 1 bis 6) weist folgende Funktionen auf:

RE(z) : Realteil (reeller Teil) einer komplexen Zahl
IM(z) : Imaginärer Teil einer komplexen Zahl
C→R(z) : Nehmen Sie eine komplexe Zahl (x,y) und trennen Sie diese in ihre reellen und imaginären Teile
R→C(x,y) : Bildet die komplexe Zahl *(x,y)* aus den reellen Zahlen x und y

ABS(z) : Berechnet die Magnitude einer komplexen Zahl oder den absoluten Wert einer reellen Zahl.
ARG(z) : Berechnet das Argument einer komplexen Zahl.

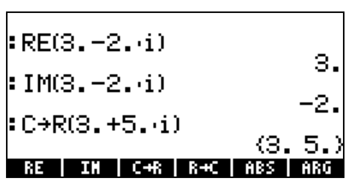
Die noch verbleibenden Optionen (Optionen 7 bis 10) sind nachfolgende:



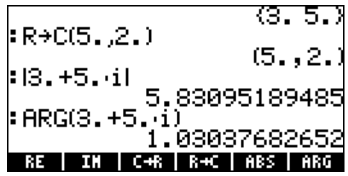
SIGN(z) : Berechnet eine komplexe Zahl der Einheit Magnitude als z/|z|.
NEG : Ändert das Vorzeichen von z
CONJ(z) : Erzeugt die konjugiert komplexe Zahl von z

Nachfolgend einige Anwendungsbeispiele dieser Funktionen. Merken Sie sich, dass im ALG-Modus das Argument der Funktion vorangestellt werden muss, während im RPN-Modus erst das Argument eingegeben, dann die Funktion ausgewählt wird. Vergessen Sie auch nicht, dass Sie diese Funktionen als Funktionsmenüs erreichen können, indem Sie das System Flag 117 setzen (siehe Kapitel 3).

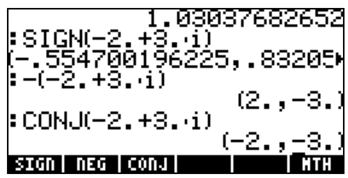
In der ersten Anzeige sind die Funktionen RE, IM und C→R zu sehen. Beachten Sie, dass diese Funktion eine Liste {3. 5.} zurückgibt, welche die reellen und imaginären Komponenten der komplexen Zahl darstellen:



In der nachfolgenden Anzeige sind die Funktionen R→C, ABS und ARG abgebildet. Beachten Sie dabei, dass die Funktion ABS als |3.+5.·i| übersetzt wird, die Notation des absoluten Wertes. Auch das Ergebnis der Funktion ARG, welche einen Winkel darstellt, wird in den zuletzt ausgewählten Winkeleinheiten ausgegeben. In unserem Beispiel wird ARG(3.+5.·i) = 1,0303… in Radian ausgegeben.

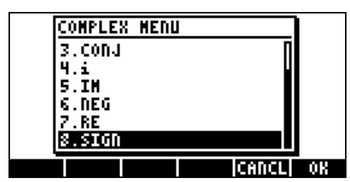


In der nächsten Abbildung stellen wir Beispiele zu den Funktionen SIGN, NEG (welche als das negative Zeichen – angezeigt wird) und CONJ dar.



Das Menü CMPLX auf der Tastatur

Ein zweites CMPLX-Menü kann über die Tastatur erreicht werden, indem Sie die rechte Shift-Taste, der Taste [/] zugeordnet, d.h. [→] CMPLX verwenden. Mit dem System Flag 117 auf CHOOSE Kästchen eingestellt, wird das CMPLX Menü wie folgt angezeigt:

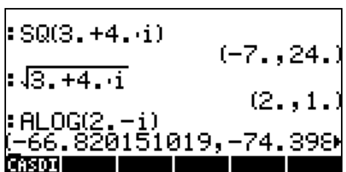


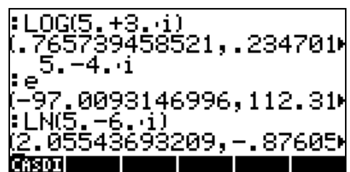
Das so erhaltene Menü enthält einige bereits im vorangegangenen Abschnitt vorgestellte Funktionen, und zwar ARG, ABS, CONJ, IM, NEG, RE und SIGN. Auch die Funktion *i* ist eingeschlossen, welche der Tastenkombination [←] *i* entspricht, d.h. um die imaginäre Einheit *i* in einen Ausdruck einzugeben.

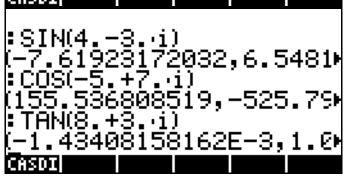
Das tastaturbasierte Menü CMPLX ist eine Alternative zum MTH-basierten Menü CMPLX, in welchem Grundfunktionen für komplexe Zahlen enthalten sind. Nehmen Sie die vorher gezeigten Beispiele, unter Verwendung des tastaturbezogenen Menüs CMPLX als Übung.

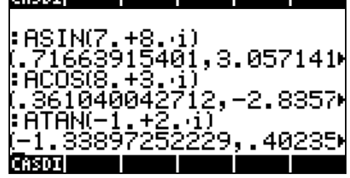
Auf komplexe Zahlen angewandte Funktionen

Viele der tastaturbasierten Funktionen für reelle Zahlen in Kapitel 3, z.B. SQ, ,LN, e^x, LOG, 10^X, SIN, COS, TAN, ASIN, ACOS, ATAN können auch auf komplexe Zahlen angewandt werden. Das Ergebnis ist eine weitere komplexe Zahl, wie in nachfolgenden Beispielen zu sehen ist. Um diese Funktionen anzuwenden, gehen Sie genau wie vorher für reelle Zahlen beschrieben (siehe Kapitel 3) vor.







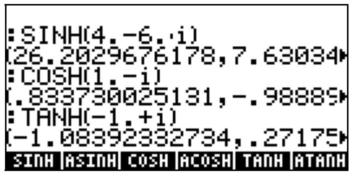


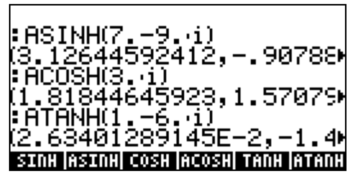
Anmerkung: Wenn Sie trigonometrische Funktionen und deren Inverse mit komplexen Zahlen verwenden, sind die Argumente keine Winkel mehr. Deshalb hat das für den Rechner ausgewählte Winkelmaß in der Berechnung dieser Funktionen mit komplexen Argumenten keine Auswirkung auf die Berechnung. Um zu verstehen wie trigonometrische und andere Funktionen für komplexe Zahlen definiert werden, lesen Sie in einem Buch über komplexe Variablen nach.

Funktionen aus dem Menü MTH

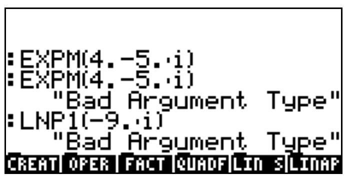
Die Hyperbelfunktionen und deren Inverse, wie auch die Gamma, PSI und Psi Funktionen (Sonderfunktionen) wurden bereits in Kapitel 3 vorgestellt und auf reelle Zahlen angewandt. Diese Funktionen können auch auf komplexe

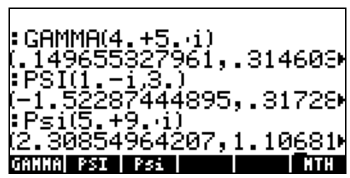
Zahlen, in der gleichen Form, wie für reelle, angewandt werden. Nachfolgend einige Beispiele:





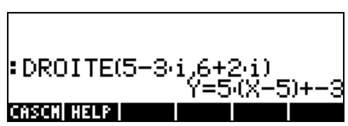
Die nachfolgende Anzeige demonstriert, dass die Funktionen EXPM und LNP1 auf komplexe Zahlen nicht angewandt werden können. Hingegen akzeptieren die Funktionen GAMMA, PSI und PSi komplexe Zahlen:





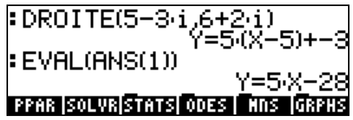
Funktion DROITE: Gleichung einer Geraden (gerade Linie)

Die Funktion DROITE hat als Argument zwei komplexe Zahlen, sagen wir x_1+iy_1 und x_2+iy_2 und gibt als Ergebnis die Gleichung einer Geraden, sagen wir $y = a+bx$, welche die Punkte (x_1,y_1) und (x_2,y_2) enthält, aus. So z.B. kann die Linie zwischen den Punkten A(5,-3) und B(6,2) wie folgt gefunden werden (Beispiel im algebraischen Modus):



Die Funktion DROITE finden Sie im Befehle Katalog (↱ CAT).

Durch Verwendung von EVAL(ANS(1)) wird das Ergebnis wie folgt vereinfacht:


:DROITE(5-3·i,6+2·i)
Y=5·(X-5)+-3
:EVAL(ANS(1))
Y=5·X-28
PPAR
SOLVR
STATS
ODES
ANS
GRPHS

Kapitel 5
Algebraische und arithmetische Operationen

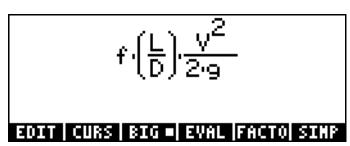
Ein algebraisches Objekt , oder einfach, Algebraik, kann jede Zahl, Variable oder algebraischer Ausdruck sein, der nach den Regeln der Algebra berechnet, manipuliert oder kombiniert werden kann. Beispiele von algebraischen Objekten sind:

- Eine Zahl: 12,3, 15,2_m, 'π', 'e', 'i'
- Der Name einer Variablen: 'a', 'ux', 'width', usw.
- Ein Ausdruck: 'p*D^2/4','f*(L/D)*(V^2/(2*g))'
- Eine Gleichung: 'Q=(Cu/n)*A(y)*R(y)^(2/3)*So^0,5'

Eingabe von algebraischen Objekten

Algebraische Objekte können mit Hilfe von einfachen Anführungszeichen (') direkt in den Stack, Ebene 1 oder über den EquationWriter (↱ EQW) eingegeben werden. Ein Beispiel wie Sie das algebraische Objekt 'π*D^2/4' direkt in den Stack, Ebene 1 eingeben können: ' ↰ π × ALPHA D y^x 2 ÷ 4 ENTER. Die resultierende Anzeige wird für den ALG-Modus links und für den RPN-Modus rechts dargestellt:

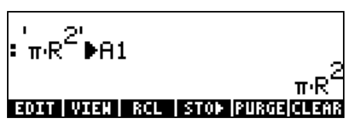
Ein algebraisches Objekt kann auch im EquationWriter erstellt und anschließend an den Stack gesandt werden. Die Bedienung des EquationWriters wurde in Kapitel 2 beschrieben. Als Übung erstellen Sie das folgende algebraische Objekt im EquationWrier:



Nachdem Sie das Objekt erzeugt haben, drücken Sie um dieses im Stack anzuzeigen (nachfolgend im ALG und RPN Modus angezeigt)

Einfache Operationen mit algebraischen Objekten

Algebraische Objekte können genau wie jede reelle oder komplexe Zahl addiert, subtrahiert, multipliziert, dividiert (ausgenommen durch Null), potenziert, als Argumente für eine Reihe von Standardfunktionen (exponential, logarithmisch, trigonometrisch, hyperbolisch usw.) verwendet werden. Um die Grundoperationen mit algebraischen Objekten zu veranschaulichen, erstellen wir einige Objekte, sagen wir ‘π*R^2’ and ‘g*t^2/4’ und speichern diese in den Variablen A1 und A2 (siehe Kapitel 2, erstellen von Variablen und speichern von Werten in denselben). Hier die Tastenfolge, um die Variable A1 im ALG-Modus einzugeben: :['] [↰] π [×] [ALPHA] [R] [Y^x] [2] [▶] [STO▶] [ALPHA] [A] [1] [ENTER], die Anzeige sieht wie folgt aus:

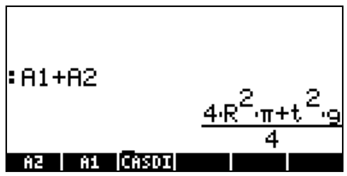


Die Tastenfolge für den RPN-Modus sieht so aus: ['] [↰] π [×] [ALPHA] [R] [Y^x] [2] [ENTER] [ALPHA] [A] [1] [STO▶]

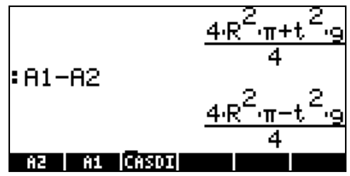
Nachdem Sie nun die Variable A2 gespeichert und die Taste gedrückt haben, erscheinen die Variablen in der Anzeige wie folgt:

Im ALG-Modus zeigen folgende Tastenanschläge eine Anzahl von Operationen mit algebraischen Zahlen, die in den Variablen A1 und A2 enthalten sind (drücken Sie VAR, um zum Variablen-Menü zurückzukehren)

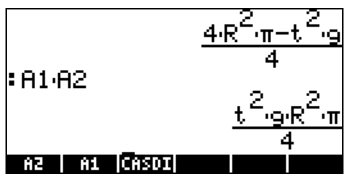
A1 + A2 ENTER



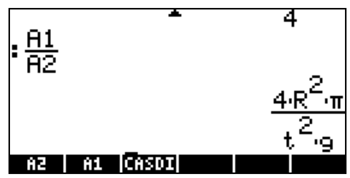
A1 − A2 ENTER



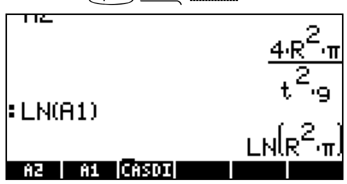
A1 × A2 ENTER



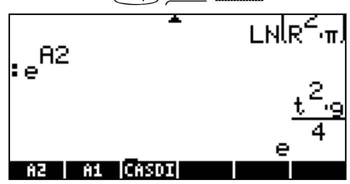
A1 ÷ A2 ENTER



→ LN A1



← e^x A2



Zum gleichen Ergebnis kommen Sie, wenn Sie im RPN-Modus die nachfolgenden Tastenfolgen verwenden:

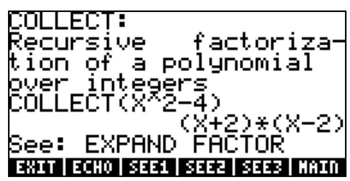
A1 ENTER A2 +	A1 ENTER A2 −
A1 ENTER A2 ×	A1 ENTER A2 ÷
A1 ENTER → LN	A2 ENTER ← e^x

Funktionen im Menü ALG

Das Menü ALG (algebraisch) erreicht man über die Tastenfolge → ALG (der Taste → zugeordnet). Mit dem System Flag 117 auf *CHOOSE* Kästchen gesetzt, zeigt das Menü ALG folgende Funktionen an:

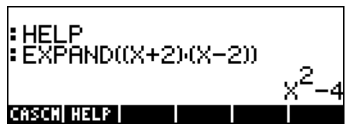
Wir wollen hier keine Beschreibung jeder einzelnen Funktion bringen, sondern den Anwender darauf hinweisen, dass er sich diese lieber Mal in der Hilfefunktion des Rechners anzeigen lassen sollte: TOOL NXT HELP ENTER . Um eine bestimmte Funktion auszuwählen, geben Sie den ersten Buchstaben der Funktion ein. So z.B. geben Sie für die Funktion COLLECT, ALPHA C ein und verwenden anschließend die Pfeiltasten ▲▼, um COLLECT im Hilfefenster zu lokalisieren.

Um den Vorgang abzuschließen, drücken Sie OK Nachfolgend die Hilfeansicht für die Funktion COLLECT:



Wir beachten, am unteren Rand der Anzeige die Zeile "See:" (siehe): EXPAND FACTOR verweist auf andere Hilfefunktionseinträge, die Funktionen EXPAND und FACTOR. Um direkt zu diesen Einträgen zu gelangen, drücken Sie die Funktionstaste SEE1 für EXPAND und SEE2 für FACTOR. Wenn Sie z.B. SEE1 drücken, erhalten Sie nachfolgende Informationen für EXPAND:

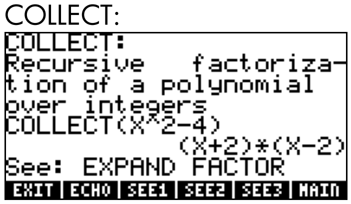
Die Hilfefunktion gibt nicht nur Informationen über die einzelnen Befehle, sondern auch ein Anwendungsbeispiel. Dieses Beispiel kann in Ihren Stack, durch drücken der Funktionstaste ECHO kopiert werden. Um z.B. für den obigen Eintrag zu EXPAND, das Beispiel in den Stack zu kopieren, drücken Sie die Funktionstaste ECHO (drücken Sie ENTER, um den Befehl auszuführen):



Nun, überlassen wir es dem Benutzer die Anwendung dieser Funktionen im ALG (oder ALGB) Menü selbst zu ergründen. Dies ist eine Liste der Befehle:

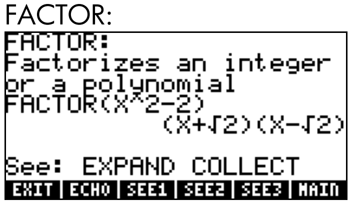
Die Hilfefunktion zeigt folgende Informationen zu den Befehlen:

COLLECT:

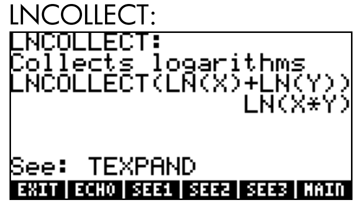


EXPAND:

FACTOR:

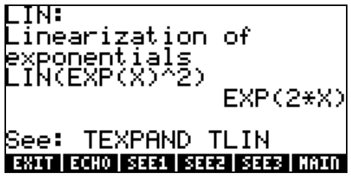


LNCOLLECT:

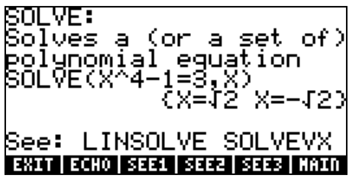


LIN:

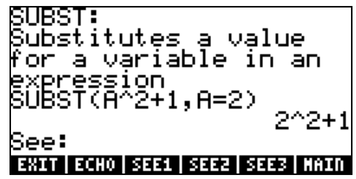
PARTFRAC:



SOLVE:



SUBST:



TEXPAND:

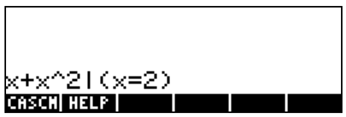
Anmerkung: Merken Sie sich: im PRN-Modus, muss das jeweilige Argument der Funktion vorangestellt werden, erst dann wird die Funktion selbst ausgewählt. So z.B. müssen Sie für TEXPAND im RPN-Modus, wie folgt vorgehen:

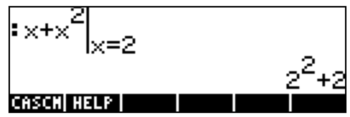
['] [↰] e^x [+] [ALPHA] [X] [+] [ALPHA] [Y] [ENTER]

Wählen Sie an dieser Stelle die Funktion TEXPAND aus dem Menü ALG (oder direkt aus dem Katalog [↱] CAT), um die Operation abzuschließen.

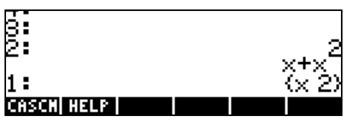
Weitere Möglichkeiten des Ersetzens in algebraischen Ausdrücken

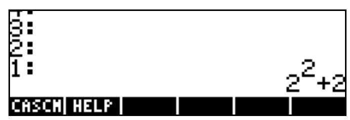
Die oben gezeigte Funktion SUBST, wird dazu verwendet, eine Variable in einem Ausdruck zu ersetzen. Eine zweite Art des Ersetzens kann mit der der Tastenfolge [↱] | (der Taste I zugeordnet) erreicht werden. Als Beispiel, wird die nachfolgende Eingabe den Wert *x = 2* im Ausdruck $x+x^2$, im ALG-Modus ersetzen. Die Zahl auf der linken Seite zeigt die Art und Weise, wie dieser Ausdruck einzugeben ist, (der ersetzte Wert, x=2, muss zwischen zwei Klammern stehen) bevor Sie die Taste ENTER betätigen. Nachdem die Taste ENTER gedrückt wurde, wird das Ergebnis in der rechten Abbildung angezeigt:





Im RPN-Modus wird dies erreicht, wenn Sie zuerst den Ausdruck, in welchem der Austausch stattfinden soll ($x+x^2$), gefolgt von einer Liste (siehe Kapitel 8) in welchem die zu ersetzende Variable enthalten ist, ein Leerschritt und der Wert, der ausgetauscht werden soll, d.h. {x 2}, eingeben. Der letzte Schritt, um den Vorgang abzuschließen, ist die Tastenkombination: ↱ | .

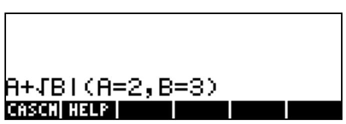


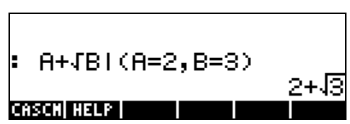


Die dazu notwendigen Tastenanschläge sind wie folgt:

' ALPHA ↰ X + ALPHA ↰ X y^x 2 ENTER
↰ {} ALPHA ↰ X SPC 2 ENTER ↱ | ENTER

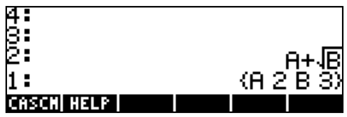
Im ALG-Modus kann mehr als eine Variable ersetzt werden, wie dies im nachfolgenden Beispiel gezeigt wird (vor und nachdem Sie die Taste ENTER gedrückt haben).

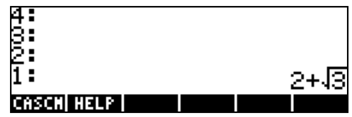




Im RPN-Modus kann ebenfalls mehr als eine Variable gleichzeitig ersetzt werden, wie in nachfolgendem Beispiel angezeigt: Beachten Sie, dass im

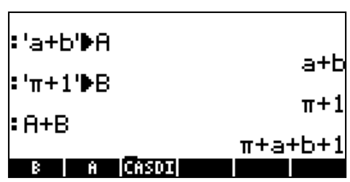
RPN-Modus eine Liste von Variablen-Namen und Werten für den Austausch verwendet wird.





Ein anderen Ansatz für den Austausch ist, die im Rechner zu ersetzenden Ausdrücke als Variablen zu definieren und die Namen der Variablen in den ursprünglichen Ausdruck einzufügen. Im ALG-Modus speichern Sie z.B. folgende Variablen:

Geben Sie anschließend den Ausdruck A+B ein:



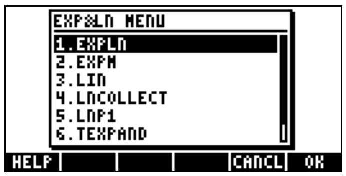
Der zuletzt eingefügte Ausdruck wird automatisch nach drücken der Taste ENTER ausgewertet und Sie erhalten das oben gezeigte Ergebnis.

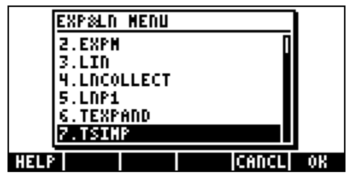
Operationen mit transzendenten Funktionen

Im Rechner stehen eine ganze Reihe von Funktionen, welche zum Austausch von Ausdrücken mit logarithmischen, Exponential-, trigonometrischen und Hyperbel-Funktionen, als trigonometrische Einheiten oder als Exponentialfunktionen, verwendet werden können, zur Verfügung. Die Menüs mit Funktionen zum Austausch von trigonometrischen Funktionen können direkt von der Tastatur aus gestartet werden, indem Sie die rechte Shift-Taste gefolgt von der Taste 8, d.h. TRIG drücken. Die Kombination dieser Taste mit der linken Shift-Taste, d.h. EXP&LN , startet ein Menü, welches Ihnen erlaubt Ausdrücke anhand von Exponential oder natürlichen Logarithmus-Funktionen zu ersetzen. In den folgenden Abschnitten werden diese Menüs im Detail vorgestellt.

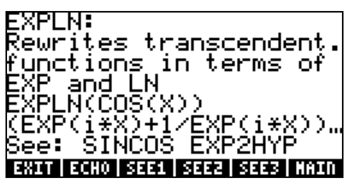
Erweitern und zusammenfassen mit Hilfe der log-exp Funktionen

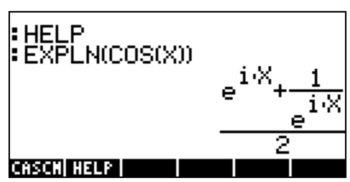
Mit EXP&LN erhalten Sie nachfolgendes Menü:





Informationen und Beispiele zu diesen Befehlen erhalten Sie über die Hilfefunktion des Rechners. Einige der Befehle aus dem Menü EXP&LN, d.h. LIN, LNCOLLECT und TEXPAND sind auch im ALG-Menü, welches vorher vorgestellt wurde, enthalten. Die Funktionen LNP1 und EXPM wurden im Menü HYPERBOLIC, unter dem Menü MTH vorgestellt (siehe Kapitel 2). Die einzige noch verbleibende Funktion ist EXPLN. Die Beschreibung dieser Funktion ist auf der linken Abbildung zu sehen, das Beispiel dazu aus der Hilfefunktion rechts:





Erweitern und zusammenfassen anhand trigonometrischer Funktionen

Das Menü TRIG wird über die Tastenkombination TRIG aufgerufen und enthält folgende Funktionen:

Mit Hilfe diesen Funktionen können Ausdrücke, durch austauschen einer bestimmten trigonometrischen Kategorie mit einer anderen, vereinfacht werden. So z.B. erlaubt die Funktion ACOS2S das ersetzen der Funktion *arccosine* (acos(x)) durch deren Ausdruck als (asin(x)).

Beschreibung dieser Befehle und Beispiele, sowie deren Anwendung finden Sie über die Hilfefunktion des Rechners (TOOL NXT HELP). Der Anwender wird dazu aufgefordert diese Hilfe nach Informationen zu den Befehlen im Menü TRIG zu suchen.

Beachten Sie, dass der erste Befehl im Menü TRIG das Menü HYPERBOLIC darstellt, dessen Funktionen in Kapitel 2 bereits vorgestellt wurden.

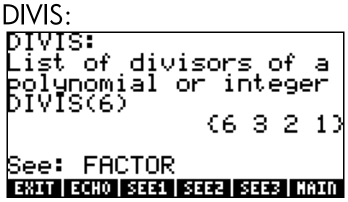
Funktionen im Menü ARITHMETIC

Das Menü ARITHMETIC enthält eine Anzahl von Unter-Menüs für spezifische Anwendungen in der Zahlentheorie (Integer, Polynome usw.), aber auch eine Anzahl von Funktionen, die für allgemeine arithmetische Operationen gültig sind. Das ARITHMETIC Menü wird über die Tastenkombination ← ARITH (der Taste 1 zugeordnet) gestartet. Mit dem System Flag 117 auf *CHOOSE* Kästchen gesetzt, erscheint über ← ARITH das nachfolgende Menü:

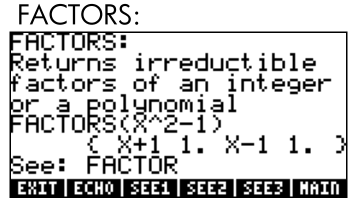
Die Optionen 5 bis 9 (*DIVIS, FACTORS, LGCD, PROPFRAC, SIMP2*) aus dieser Liste, entsprechen den allgemeinen Funktionen für Ganzzahlen und Polynome. Die verbliebenen Optionen (*1. INTEGER*, *2. POLYNOMIAL*, *3. MODULO*, and *4. PERMUTATION*) sind eigentlich Untermenüs von Funktionen, welche bestimmten mathematischen Objekten zugeordnet sind. Der Unterschied zwischen den Untermenüs (Optionen 1 bis 4) und reinen Funktionen (Optione 5 bis 9) wird klar, wenn das System Flag 117 auf *SOFT* Menüs gesetzt ist. Starten Sie das Menü ARITHMETIC (ARITH) unter diesen Umständen, erhalten Sie:

Nachfolgend stellen wir Ihnen die Hilfefunktion für die Einträge der Funktionen aus den Optionen 5 bis 9 aus dem Menü ARITHMETIC vor:

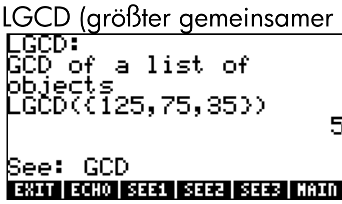
DIVIS:



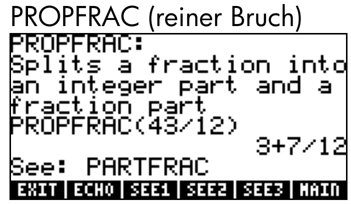
FACTORS:



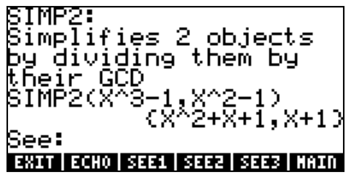
LGCD (größter gemeinsamer Nenner):



PROPFRAC (reiner Bruch)



SIMP2:



Die den ARITHMETIC Unter-Menüs zugeordneten Funktionen: INTEGER, POLYNOMIAL, MODULO und PERMUTATION sind wie folgt:

Menü INTEGER

EULER	Integerzahlen < n, koprim/teilerfremd mit n
IABCUV	Löst au + bv = c, wobei a,b,c = Integerzahlen sind
IBERNOULLI	n-te Bernoulli Zahl
ICHINREM	Chinesischer Restesatz für Integerzahlen
IDIV2	Euklidische Division von zwei Integerwerten
IEGCD	Gibt als Ergebnis u,v, so dass au + bv = gcd(a,b)
IQUOT	Euklidischer Quotient zweier Integerwerte
IREMAINDER	Euklidischer Algorithmus zweier Integerwerte mit Rest
ISPRIME?	Überprüft, ob eine Integerzahl eine Primzahl ist
NEXTPRIME	Nächste Primärzahl für eine vorgegebene Integerzahl
PA2B2	Primärzahl als Quadratnorm einer komplexen Zahl
PREVPRIME	Vorhergehende Primärzahl für eine vorgegebene Integerzahl

Menü POLYNOMIAL (Polynome)

ABCUV	Polynomengleichung nach Bézout (au+bv=c)
CHINREM	Chinesischer Restesatz für Polynome
CYCLOTOMIC	n-tes Kreisteilungs-Polynom
DIV2	Euklidische Division zweier Polynome
EGDC	Gibt u,v aus au+bv=gcd(a,b) zurück
FACTOR	Zerlegt eine Integerzahl oder ein Polynom in Faktoren
FCOEF	Erzeugt einen Bruch bei gegebenen Wurzeln und Mehrwertigkeit
FROOTS	Gibt Wurzel und Mehrwertigkeit eines Bruches zurück

GCD	Größter gemeinsamer Teiler zweier Zahlen eines Polynoms
HERMITE	Hermite-Polynom n-ten Grades
HORNER	Hornersche Berechnung eines Polynoms
LAGRANGE	Lagrange-Interpolation des Polynoms
LCM	Kleinstes gemeinsames Vielfaches zweier Zahlen oder Polynome
LEGENDRE	Legendre-Polynom n-ten Grades
PARTFRAC	Partialbruch-Zerlegung eines gegebenen Bruches
PCOEF	(Hilfefunktionseintrag fehlt)
PTAYL	Gibt Q(x-a) in Q(x-a) = P(x) zurück, Taylor Polynom
QUOT	Euklidischer Quotient zweier Polynome
RESULTANT	Determinante der Sylvester-Matrix zweier Polynome
REMAINDER	Euklidischer Restesatz zweier Polynome
STURM	Sturm-Kette eines Polynoms
STURMAB	Zeichen an unterer Grenze und Anzahl der Nullen zwischen den Grenzen

Menü MODULO

ADDTMOD	Addition zweier Ausdrücke Modulo aktuelles Modul (absoluter Betrag)
DIVMOD	Division 2 Polynome Modulo aktuelles Modul (absoluter Betrag)
DIV2MOD	Euklidische Division zweier Polynome mit Modulo Koeffizienten
EXPANDMOD	Erweitert/vereinfacht Polynome Modulo, aktuelles Modul (absoluter Betrag)
FACTORMOD	zerlegt ein Polynome Modulo in Faktoren, aktuelles Modul (absoluter Betrag)
GCDMOD GCD	(größter gemeinsamer Teiler) zweier Polynome Modulo aktuelles Modul (absoluter Betrag)
INVMOD	Inverse der Integer Modulo, aktuelles Modul (absoluter Betrag)
MOD	(kein Eintrag in der Hilfefunktion vorhanden)
MODSTO	Ändert die Modulo Einstellung auf den angegebenen Wert

MULTMOD	Multiplikation zweier Polynome Modulo aktuelles Modul (absoluter Betrag)
POWMOD	Potenziert ein Polynom auf einen Exponenten Modulo, aktuelles Modul (absoluter Betrag)
SUBTMOD	Subtraktion zweier Polynome Modulo aktuelles Modul (absoluter Betrag)

Anwendungen des Menüs ARITHMETIC

In diesem Abschnitt werden Hintergrundinformationen für die Anwendung der Funktionen des Menüs ARITHMETIC dargestellt. Gleich anschließend werden Definitionen zu den Subjekten von Polynomen, polynomischen Brüchen und zur modularen Arithmetik erläutert. Die vorgestellten Beispiele werden unabhängig von den Einstellungen des Rechners (ALG oder RPN) vorgestellt.

Modulare Arithmetik

Nehmen wir ein Zahlensystem bestehend aus Integerzahlen, welche periodisch auf sich selbst zurückgehen und neu starten, wie die Stunden einer Uhr. Ein solches Zählsystem wird als *Ring* bezeichnet. Da die in einem Ring verwendete Anzahl von Integerwerten begrenzt ist, wird die Arithmetik in diesem Ring als endliche Arithmetik bezeichnet. Nehmen wir an unsere endliche Zahl an Integerwerten besteht aus den Zahlen *0, 1, 2, 3, …, n-1, n*. Die Arithmetik in diesem Zählsystem können wir auch als *modulare Arithmetik des Moduls n* bezeichnen. Im Falle der Stunden einer Uhr, wäre das Modul 12. (Wenn wir aber in der modularen Arithmetik mit den Stunden einer Uhr arbeiten, müssten wir die Integerzahlen 0, 1, 2, 3, …, 10, 11, und nicht 1, 2, 3,…,11, 12) benutzen.

Operationen in modularer Arithmetik

Addition in modularer Arithmetik mit dem Modul *n*, welches eine positive Integerzahl darstellt, wenn *j* und *k* zwei positive Integerzahlen, beide kleiner als *n* sind, und wenn $j+k\geq n$, dann wird $j+k$ als $j+k-n$ definiert Im Beispiel mit unserer Uhr wäre das, für *n* = 12, 6+9 "=" 3. Um diese "Gleichwertigkeit" von unendlichen arithmetischen Gleichheiten zu unterscheiden, wird das Symbol ≡ anstelle des Istgleichzeichens gesetzt und das Verhältnis zwischen diesen Zahlen als *Kongruenz*und nicht als Gleichwertigkeit bezeichnet. Somit würden wir für das obige Beispiel $6+9 \equiv 3 \pmod{12}$ schreiben und diesen

Ausdruck wie folgt lesen "*sechs plus neun ist kongruent zu drei, Modul 12*". Stellen die Zahlen die Stunden seit Mitternacht dar, kann z.B. die Kongruenz 6+9 ≡ 3 (mod 12) als "sechs Stunden nach der neunten Stunde nach Mitternacht, wird drei Stunden nach Mittag sein" interpretieren. Andere Summen, welche im Modul 12 Arithmetik definiert werden können sind: 2+5 ≡ 7 (mod 12); 2+10 ≡ 0 (mod 12); 7+5 ≡ 0 (mod 12); und so weiter.

Die Regeln für die Subtraktion verhalten sich so: wenn $j – k < 0$, dann wird j-k als j-k+n definiert. Somit, liest man *8-10 ≡ 2 (mod 12)* als "*acht minus zehn ist kongruent zu zwei, Modul zwölf*". Ein weiteres Beispiel einer Subtraktion in Modul 12 Arithmetik wäre 10-5 ≡ 5 (mod 12); 6-9 ≡ 9 (mod 12); 5 – 8 ≡ 9 (mod 12); 5 –10 ≡ 7 (mod 12); und so weiter.

Die *Multiplikation* erfolgt nach der Regel, dass wenn $j \cdot k > n$, wobei $j \cdot k = m \cdot n + r$, wobei m und r positive Integerzahlen sind, beide kleiner als n, dann ist $j \cdot k \equiv r$ (mod n). Das Multiplikationsprodukt von j mal k in Modul n Arithmetik ist im Grunde genommen der Ganzzahlrest von $j \cdot k/n$ in der unendlichen Arithmetik, wenn $j \cdot k>n$. So z.B. im Modul 12 Arithmetik haben wir 7·3 = 21 = 12 + 9, (or, 7·3/12 = 21/12 = 1 + 9/12, d.h., der Ganzzahlrest von 21/12 is 9) Wir können nun 7·3 ≡ 9 (mod 12) schreiben und das letzte Ergebnis wie folgt lesen "*sieben mal drei ist kongruent zu neun, Modul zwölf*".

Der *Divisionsvorgang* kann als Multiplikation wie folgt ausgeführt werden $r/k \equiv j$ (mod n), if, $j \cdot k \equiv r$ (mod n). Das bedeutet, dass r den Restwert von $j \cdot k/n$ darstellen muss. z.B. 9/7 ≡ 3 (mod 12), weil 7·3 ≡ 9 (mod 12) darstellt. Einige Divisionen sind in der modularen Arithmetik nicht erlaubt. So z.B. in Modul 12 Arithmetik können Sie 5/6 (mod 12) nicht definieren, weil die Multiplikationstabelle von 6, das Ergebnis 5 nicht im Modul 12 Arithmetik anzeigt. Nachfolgend die Multiplikationstabelle:

6*0 (mod 12)	0	6*6 (mod 12)	0
6*1 (mod 12)	6	6*7 (mod 12)	6
6*2 (mod 12)	0	6*8 (mod 12)	0
6*3 (mod 12)	6	6*9 (mod 12)	6
6*4 (mod 12)	0	6*10 (mod 12)	0
6*5 (mod 12)	6	6*11 (mod 12)	6

Formdefinition eines endlichen arithmetischen Ringes

Der Ausdruck $a \equiv b \ (mod\ n)$ wird als "a ist kongruent zu b, Modulo n" interpretiert und wenn *(b-a)* ein Vielfaches von *n* beinhaltet. Anhand dieser Definition werden die arithmetischen Regeln wie folgt vereinfacht:

Wenn $a \equiv b \ (mod\ n)$ und $c \equiv d \ (mod\ n)$,

dann

$$a+c \equiv b+d \ (mod\ n),$$
$$a-c \equiv b - d \ (mod\ n),$$
$$a \times c \equiv b \times d \ (mod\ n).$$

Für die Division befolgen Sie die vorher beschriebenen Regeln. Z.B. ist $17 \equiv 5$ (mod 6) und $21 \equiv 3$ (mod 6). Unter Verwendung dieser Regeln können wir schreiben:

$17 + 21 \equiv 5 + 3$ (mod 6) => $38 \equiv 8$ (mod 6) => $38 \equiv 2$ (mod 6)
$17 - 21 \equiv 5 - 3$ (mod 6) => $-4 \equiv 2$ (mod 6)
$17 \times 21 \equiv 5 \times 3$ (mod 6) => $357 \equiv 15$ (mod 6) => $357 \equiv 3$ (mod 6)

Beachten Sie, dass jedes Mal wenn das Ergebnis auf der rechten Seite der "Kongruenz" größer als das Modulo ist, (in diesem Fall n = 6), können Sie immer ein Vielfaches des Modulo von diesem Ergebnis abziehen und zu einer Nummer, die kleiner als das Modulo ist vereinfachen. Somit können die Ergebnisse aus dem ersten Fall *8 (mod 6)* auf *2 (mod 6)* vereinfacht werden und die Ergebnisse im dritten Fall, *15 (mod 6)* auf *3 (mod 6).* Verwirrend? Eigentlich nicht, wenn Sie den Rechner diese Operationen selbst durchführen lassen. Lesen Sie den nächsten Abschnitt, zum besseren Verständnis, wie Sie mit endlichen arithmetischen Ringen in Ihrem Rechner umzugehen haben.

Endliche arithmetische Ringe im Rechner

Schon immer haben wir unsere endlichen arithmetischen Operationen so definiert, dass deren Ergebnisse einen positiven Wert ergeben. Das in Ihrem Rechner existierende arithmetische System ist so eingestellt, dass der Modul Ring n die Zahlen $-n/2+1, \ldots,-1, 0, 1,\ldots,n/2-1, n/2$, wenn n eine Paarzahl ist und *–(n-1)/2, -(n-3)/2,…,-1,0,1,…,(n-3)/2, (n-1)/2* einschließt, wenn n eine ungerade Zahl darstellt. Für n = 8 (gerade) schließt der arithmetische Ring Ihres Rechners z.B. die Zahlen (-3,-2,-1,0,1,3,4) ein, während für n = 7 (ungerade) der entsprechende endliche arithmetische Ring durch die Zahlen (-3,-2,-1,0,1,2,3) dargestellt wird.

Modulare Arithmetik im Rechner

Um das modulare arithmetische Menü im Rechner zu starten, wählen Sie das Untermenü MODULO aus dem Menü ARITHMETIC (← ARITH). Das vorhandene Menü beinhaltet die Funktionen: ADDTMOD, DIVMOD, DIV2MOD, EXPANDMOD, FACTORMOD, GCDMOD, INVMOD, MOD, MODSTO, MULTMOD, POWMOD, and SUBTMOD. Eine kurze Beschreibung dieser Funktionen finden Sie in einem früheren Abschnitt. Als nächstes stellen wir die Anwendung dieser Funktionen vor.

Einstellung des Moduls (oder MODULO)

Im Rechner befindet sich eine Variable mit dem Namen MODULO welche sich im Verzeichnis {HOME CASDIR} befindet und in welcher die Magnitude oder das MODULO für modulare arithmetische Anwendungen gespeichert wird.

Der Standartwert für MODULO ist 13. Um den Wert von MODULO zu ändern, können Sie entweder den neuen Betrag direkt in die Variable MODULO ins Unterverzeichnis {HOME CASDIR} speichern. Alternativ dazu können Sie einen neuen MODULO Wert über die Funktion MODSTO speichern.

Modulare arithmetische Operationen mit Zahlen

Verwenden Sie die Funktionen ADDTMOD, SUBTMOD, MULTMOD, DIV2MOD und DIVMOD (für Division) und POWMOD, zur Addition, Subtraktion, Multiplikation, Division oder Potenzierung in der modularen Arithmetik. Im RPN-Modus müssen Sie zwei Zahlen für die Operation eingeben, beide durch ein [ENTER] oder ein [SPC] voneinander getrennt, drücken Sie anschließend die entsprechende modulare arithmetische Funktion. Wollen Sie z.B. Modul von 12 verwenden, versuchen Sie folgende Operationen:

ADDTMOD Beispiel

6+5 ≡ -1 (mod 12)	6+6 ≡ 0 (mod 12)	6+7 ≡ 1 (mod 12)
11+5 ≡ 4 (mod 12)	8+10 ≡ -6 (mod 12)	

SUBTMOD Beispiele

5 - 7 ≡ -2 (mod 12)	8 – 4 ≡ 4 (mod 12)	5 –10 ≡ -5 (mod 12)
11 – 8 ≡ 3 (mod 12)	8 - 12 ≡ -4 (mod 12)	

MULTMOD Beispiele

6·8 ≡ 0 (mod 12)	9·8 ≡ 0 (mod 12)	3·2 ≡ 6 (mod 12)
5·6 ≡ 6 (mod 12)	11·3 ≡ -3 (mod 12)	

DIVMOD Beispiele

12/3 ≡ 4 (mod 12)	12/8 (mod 12) existiert nicht
25/5 ≡ 5 (mod 12)	64/13 ≡ 4 (mod 12)
66/6 ≡ -1 (mod 12)	

DIV2MOD Beispiele

2/3 (mod 12) existiert nicht
26/12 (mod 12) existiert nicht
125/17 (mod 12) ≡ 1 mit Restwert = 0
68/7 ≡ -4 (mod 12) mit Restwert = 0
7/5 ≡ -1 (mod 12) mit Restwert = 0

Anmerkung: DIVMOD ermittelt den Quotienten der modularen Division j/k (mod n), während DIMV2MOD nicht nur den Quotienten, sondern auch den Restwert der modularen Division j/d (mod n) ermittelt.

POWMOD Beispiele

2^3≡ -4 (mod 12)	3^5≡ 3 (mod 12)	5^{10}≡ 1 (mod 12)
11^8 ≡ 1 (mod 12)	6^2 ≡ 0 (mod 12)	9^9 ≡ -3 (mod 12)

In den oben gezeigten Beispielen mit modularen Arithmetik-Operationen, haben wir Zahlen benutzt, die nicht unbedingt zum Ring gehören, d.h. Zahlen wie 66, 125, 17 usw. Der Rechner wird diese Zahlen erst in Ringwerte konvertieren, erst dann die Operationen auf diese anwenden. Sie können selbst jede Zahl mit der Funktion EXPANDMOD in einen Ringwert konvertieren So zum Beispiel:

EXPANDMOD(125) ≡ 5 (mod 12)
EXPANDMOD(17) ≡ 5 (mod 12)
EXPANDMOD(6) ≡ 6 (mod 12)

Die modulare Inverse einer Zahl

Nehmen wir an eine Zahl *k* gehört einen endlichen arithmetischen Ring des Moduls *n* an, dann ist die modulare Inverse von k, d.h. *1/k (mod n)* eine Zahl *j*, die sich als $j \cdot k \equiv 1$ (mod *n*) verhält. Die modulare Inverse einer Zahl erhält man mit der Funktion INVMOD in MODULO im Untermenü des Menüs ARITHMETIC. Im Modul 12 Arithmetik z.B.:

1/6 (mod 12) existiert nicht
1/5 ≡ 5 (mod 12)
1/7 ≡ -5 (mod 12)
1/3 (mod 12) existiert nicht
1/11 ≡ -1 (mod 12)

Der MOD Operator

Der MOD Operator wird zur Ermittlung der Ringzahl, eines gegebenen Moduls, entsprechend einer gegebenen Integerzahl, verwendet. Auf Papier wird diese Operation als *m* mod *n* = *p* geschrieben und wir "*m* Modul von *n* ist gleich *p*" gelesen. So z.B. um 15 mod 8 zu berechnen, geben Sie ein:

- im ALG-Modus: 1 5 MOD 8 ENTER
- im RPN-Modus: 1 5 ENTER 8 ENTER MOD

Das Ergebnis ist 7, d.h. 15 mod 8 = 7. Versuchen Sie folgende Übungen:

18 mod 11 = 7	23 mod 2 = 1	40 mod 13 = 1
23 mod 17 = 6	34 mod 6 = 4	

Eine praktische Anwendung der Funktion MOD für Programmierzwecke ist, herauszufinden, ob eine Integerzahl gerade oder ungerade ist, da *n* mod 2 = 0, wenn *n* gerade ist ergibt und *n* mod 2 = 1, wenn *n* ungerade ist. Sie kann auch zur Ermittlung, ob *m* ein Vielfaches einer anderen Intergerzahl *n* ist; dies ist der Fall, wenn *m* mod *n* = 0 ist.

Anmerkung: Sehen Sie in der Hilfefunktion des Rechners für Beschreibung und Beispiele weiterer modulare Arithmetik nach. Viele dieser Funktionen können auf Polynome angewandt werden. Informationen zur modulare Arithmetik mit Polynomen finden in Büchern über Zahlentheorien.

Polynome

Polynome sind algebraische Ausdrücke, die aus einem oder mehreren Gliedern, welche abfallende Potenzen einer gegebenen Variable enthalten. So ist z.B. der Ausdruck 'X^3+2*X^2-3*X+2' ein Polynom dritten Grades der Variablen X, während 'SIN(X)^2-2' ein Polynome zweiten Grades in SIN(X) darstellt. Eine Aufzählung von Funktionen zu Polynomen im Menü ARITHMETIC wurde vorhin dargestellt. Als nächstes finden Sie einige allgemeine Definitionen zu Polynomen. In diesen Definitionen stellen *A(X), B(X), C(X), P(X), Q(X), U(X), V(X) usw. Polynome dar.*

- Polynombruch: ein Bruch des Zähler und Nenner Polynome sind, sagen wir C(X) = A(X)/B(X)
- Wurzeln oder Nullen eines Polynoms: Werte von X für welche P(X) = 0
- Pole eines Bruches: Wurzeln des Nenners
- Mehrwertigkeit der Wurzeln oder Pole: die Anzahl, wie of eine Wurzel auftaucht, z.B. $P(X) = (X+1)^2(X-3)$ hat die Wurzeln {-1, 3} mit den Mehrwertigkeiten {2,1}
- Kreisteilungs-Polynom ($P_n(X)$): ein Polynom EULER(*n)* Grades, dessen Wurzeln die primitive n-te Wurzel der Einheit ist, z.B. $P_2(X) = X+1$, $P_4(X) = X^2+1$
- Bézout's Polynomgleichung: A(X) U(X) + B(X)V(X) = C(X)

Spezifische Anwendungsbeispiele von Polynomen finden nachstehend.

Modulare Arithmetik mit Polynomen

Auf die gleiche Art, wie wir einen endlichen arithmetischen Ring für Zahlen in einem vorangegangenen Abschnitt definiert haben, können wir auch einen endlichen arithmetischen Ring für Polynome mit einem gegebenen Polynom als

Modul definieren. So können wir z.B. ein bestimmtes Polynom *P(X)* als *P(X) = X (mod* X^2*) definieren* oder ein anderen Polynom als *Q(X) = X + 1 (mod X-2).*

Ein Polynom *P(X)* ist Teil eines arithmetischen Ringes von Polynomen des Moduls *M(X),* wenn es ein drittes Polynom *Q(X)* gibt, und zwar so, dass *(P(X) – Q(X))* ein Vielfaches von *M(X)* darstellt. Dann würden wir schreiben: *P(X)* $\equiv$ *Q(X) (mod M(X)).* Letzterer Ausdruck wird als "*P(X) ist kongruent mit Q(X), modulo M(X)"* interpretieren.

Die Funktion CHINREM

CHINREM steht für CHINese REMainder (Chinesischer Restesatz). Die in diesem Befehl kodierte Operation, löst ein System von von Kongruenten unter Anwendung des Chinesischen Restesatz Theorems . Dieser Befehl kann mit Polynomen, wie auch mit Integerzahlen (Funktion ICHINREM) verwendet werden. Die Eingabe besteht aus zwei Vektoren *[expression_1, modulo_1] und [expression_2, modulo_2] – (Ausdruck_2, Modul_2) und (Ausdruck_2, Modul_2).* Die Ausgabe ist ein Vektor der *[expression_3, modulo_3]*, wobei *modulo_3* mit der Produkt verwandt ist *(modulo_1)·(modulo_2)* . [*(Ausdruck_3, Modul_3) und (Modul_1)· (Modul_2)*]. Beispiel: CHINREM(['X+1', 'X^2-1'],['X+1','X^2']) = ['X+1',-(X^4-X^2)]

Ansatz für das Chinesische Restsatz Theorem für Integerzahlen

Wenn $m_1, m_2, \ldots, m_r$ paarweise teilerfremde natürliche Zahlen und $a_1, a_2, \ldots, a_r$ irgendwelche Integerzahlen sind, dann gibt es genau eine Integer x, welche simultan die folgenden Kongruenzen erfüllt: $x \equiv a_1$ (mod m_1), $x \equiv a_2$ (mod m_2), …, $x \equiv a_r$ (mod m_r). Zusätzlich, wenn x = a eine Lösung ist, dann sind alle anderen Lösungen kongruent und gleich dem Produkt von $m_1 \cdot m_2 \cdot \ldots m_r$.

Die Funktion EGCD

EGCD steht für Extended Greatest Common Divisor (erweiterter (Euklidischer Algorithmus) größter gemeinsamer Teiler). Für zwei Polynome A(X) und B(X), erzeugt die Funktion EGCD die Polynome C(X), U(X), and V(X), so dass C(X) = U(X)*A(X) + V(X)*B(X). So z.B. für A(X) = X^2+1, B(X) = X^2-1, EGCD(A(X),B(X)) = {2, 1, -1}. d.h., 2 = 1*(X^2+1')-1*(X^2-1). Auch

EGCD('X^3-2*X+5','X') = { 5, '-(X^2-2)', 1}, d.h. 5 = – (X^2-2)*X + 1*(X^3-2*X+5).

Die Funktion GCD

Die Funktion GCD (Greatest Common Denominator – größter gemeinsamer Nenner) dient zur Ermittlung des größten gemeinsamen Nenners zweier Polynome oder zweier Listen von Polynomen der selben Länge. Bevor Sie die Funktion GCD anwenden, müssen die beiden Polynome oder Listen von Polynomen in Stack Ebene 2 verschoben werden. Das Ergebnis ist ein Polynom oder eine Liste, die den größten gemeinsamen Nenner der beiden Polynome oder von jeder Liste der Polynome darstellt. Es folgen Beispiele im RPN-Modus (Rechner steht im Exact Modus):

'X^3-1' ENTER 'X^2-1' ENTER GCD ergibt: 'X-1'

{'X^2+2*X+1','X^3+X^2'} ENTER {'X^3+1','X^2+1'} ENTER GCD ergibt {'X+1' 1}

Die Funktion HERMITE

Die Funktion HERMITE [HERMI] verwendet als Argument eine Integerzahl, k und gibt das Hermite Polynom k-ten Grades zurück. Ein Hermites-Polynom, $He_k(x)$ wird definiert als

$$He_0 = 1, \quad He_n(x) = (-1)^n e^{x^2/2} \frac{d^n}{dx^n}(e^{-x^2/2}), \quad n = 1,2,...$$

Eine alternative Definition eines Hermiten-Polynoms ist

$$H_0{}^* = 1, \quad H_n{}^*(x) = (-1)^n e^{x^2} \frac{d^n}{dx^n}(e^{-x^2}), \quad n = 1,2,...$$

wobei d^n/dx^n = n-te Ableitungsfunktion zu x darstellt. Dies ist die im Rechner verwendete Definition.

Beispiele: Die Hermiten Polynome dritten und fünften Grades sind wie folgt:

HERMITE(3) = '8*X^3-12*X' ,

und HERMITE(5) = '32*x^5-160*X^3+120*X'.

Die Funktion HORNER

Die Funktion HORNER erzeugt die Horner oder synthetische Division eines Polynoms P(X) durch den Faktor (X-*a*). Als Eingabe der Funktion wird das Polynome P(X) und die Zahl a benötigt. Die Funktion gibt den Quotienten des Polynoms Q(X), welcher aus der Division von P(X) durch (X-*a*) entsteht, den Wert *a* und den Wert von P(*a*) – in dieser Reihenfolge – zurück. Mit anderen Worten P(X) = Q(X)(X-a)+P(a). Zum Beispiel: HORNER('X^3+2*X^2-3*X+1',2) = {'X^2+4*X+5', 2, 11}. Wir könnten somit schreibn, dass $X^3+2X^2-3X+1 = (X^2+4X+5)(X-2)+11$. Ein weiteres Beispiel: HORNER('X^6-1',-5)= {'X^5-5*X^4+25*X^3-125*X^2+625*X-3125',-5, 15624} d.h., $X^6-1 = (X^5-5*X^4+25X^3-125X^2+625X-3125)(X+5)+15624$.

Die Variable VX

Im Verzeichnis {HOME CASDIR} gibt es eine Variable mit dem Namen VX , welche standardmäßig den Wert 'X' annimmt. Dies ist der Name der bevorzugten unabhängigen Variablen für algebraische und Calculus Anwendungen. Vermeiden den Variablennamen VX in Ihren Programmen oder Gleichungen, um eine Verwechslung mit der CAS-Variablen VX zu vermeiden. Wenn Sie sich aber auf die x-Komponente der Geschwindigkeit beziehen wollen, können Sie dafür entweder vx oder Vx benutzen. Zusätzliche Informationen zur CAS Variablen finden Sie in Anhang C.

Die Funktion LAGRANGE

Die Funktion LAGRANGE benötigt als Eingabe eine Matrix mit zwe Reihen und *n* Zeilen. Die Matrix speichert Datenpunkte in Form von [[x_1,x_2, …, x_n] [y_1, y_2, …, y_n]]. Die Funktion LAGRANGE erzeugt ein erweitertes Polynom aus

$$p_{n-1}(x) = \sum_{j=1}^{n} \frac{\prod_{k=1,k\neq j}^{n}(x-x_k)}{\prod_{k=1,k\neq j}^{n}(x_j-x_k)} \cdot y_j .$$

So z.B. für n = 2, können wir schreiben:

$$p_1(x) = \frac{x - x_2}{x_1 - x_2} \cdot y_1 + \frac{x - x_1}{x_2 - x_1} \cdot y_2 = \frac{(y_1 - y_2) \cdot x + (y_2 \cdot x_1 - y_1 \cdot x_2)}{x_1 - x_2}$$

Überprüfen Sie dieses Ergebnis mit Ihrem Rechner:
LAGRANGE([[x1,x2],[y1,y2]]) = '((y1-y2)*X+(y2*x1-y1*x2))/(x1-x2)'.

Weitere Beispiele: LAGRANGE([[1, 2, 3][2, 8, 15]]) = '(X^2+9*X-6)/2'
LAGRANGE([[0.5,1.5,2.5,3.5,4.5][12.2,13.5,19.2,27.3,32.5]]) =
'-(.1375*X^4+ -.7666666666667*X^3+ - .74375*X^2 =
1.991666666667*X-12.92265625)'.

Anmerkung: Matrizen werden in Kapitel 10 eingeführt..

Die Funktion LCM

Die Funktion LCM (Least Common Multiple – kleinstes gemeinsames Vielfaches) berechnet das kleinste gemeinsame Vielfache zweier Polynome oder Listen von Polynomen der gleichen Länge. Beispiele:

LCM('2*X^2+4*X+2' ,'X^2-1') = '(2*X^2+4*X+2)*(X-1)'.
LCM('X^3-1','X^2+2*X') = '(X^3-1)*(X^2+2*X)'

Die Funktion LEGENDRE

Ein Legendre-Polynom n-ten Grades ist eine Polynom-Funktion die die Differentialgleichung $(1 - x^2) \cdot \frac{d^2 y}{dx^2} - 2 \cdot x \cdot \frac{dy}{dx} + n \cdot (n+1) \cdot y = 0$ löst.

Um den n-ten Grad des Legendre Polynoms zu erhalten, verwenden Sie LEGENDRE(*n*), wie z.B.

LEGENDRE(3) = '(5*X^3-3*X)/2'
LEGENDRE(5) = '(63*X ^5-70*X^3+15*X)/8'

Die Funktion PCOEF

Wenn wir ein Array mit den Wurzeln des Polynoms haben, erzeugt die Funktion PCOEF ein Array, welches die Koeffizienten der entsprechenden Polynome enthält. Die Koeffizienten entsprechen in abfallender Reihenfolge des Grades der unabhängigen Variablen. So zum Beispiel: PCOEF([-2,–1,0,1,1,2]) = [1. –1. –5. 5. 4. –4. 0.], welche das Polynom X^6-X^5-$5X^4$+$5X^3$+$4X^2$-4X darstellt.

Die Funktion PROOT

Wenn wir ein Array der Koeffizienten eines Polynoms in abfallender Reihenfolge haben, erzeugt die Funktion PROOT die Wurzeln des Polynoms. Beispiel, aus X^2+5X-6 =0, PROOT([1 –5 6]) = [2. 3.].

Die Funktion PTAYL

Wenn wir ein Polynom P(X) und eine Zahl *a* haben, ergibt die Funktion PTAYL einen Ausdruck Q(X-*a*) = P(X), d.h. es erzeugt ein Polynom als Potenz von (X-*a*). Dies ist auch als Taylor-Polynom bekannt, von welchem auch der Name der Funktion abgeleitet wurde, Polynom & TAYLor, folgen.

So z.B. ergibt PTAYL('X^3-2*X+2',2) = 'X^3+6*X^2+10*X+6'.

Eigentlich sollten Sie dieses Ergebnis so interpretieren

'(X-2) ^3+6*(X-2) ^2+10*(X-2) +6'.

Überprüfen wir das, indem wir die Substitution benutzen: 'X = x – 2'. Wir stellen das ursprüngliche Polynom wieder her, aber als kleingeschriebenes x anstelle eines großgeschriebenen.

Die Funktionen QUOT und REMAINDER

Die Funktionen QUOT und REMAINDER geben entsprechend den Quotienten (Q(X) und den Restwert R(X) die aus der Division der zwei Polynome P_1(X) und P_2(X) resultieren. Mit anderen Worten, geben Sie die Werte aus Q(X) und R(X) aus P_1(X)/P_2(X) = Q(X) + R(X)/P_2(X). So zum Beispiel:

QUOT(X^3-2*X+2, X-1) = X^2+X-1
REMAINDER(X^3-2*X+2, X-1) = 1.

Somit können wir schreiben: (X^3-2X+2)/(X-1) = X^2+X-1 + 1/(X-1).

Anmerkung: Sie könnten letzteres Ergebnis auch üfr PROPFRAC erhalten: PROPFRAC(‘(X^3-2*X+2)/(X-1)’) = ‘X^2+X-1 + 1/(X-1)’.

Die Funktion EPSX0 und die CAS Variable EPS

Die Variable ε (Epsilon) wird normalerweise in mathematischen Textbüchern, zur Darstellung einer sehr kleinen Zahl, verwendet. Verwenden Sie die Funktion EPSX0 wird im CAS des Rechners eine Variable EPS mit dem Standardwert 0,0000000001 = 10^{-10} erzeugt. Sobald diese Variable erzeugt wurde, können Sie den Wert dieser, in einen von Ihnen gewünschten Wert für EPS. ändern. Wird die Funktion EPSX0 auf ein Polynom angewendet, werden alle Koeffizienten deren absoluter Wert kleiner als EPS ist mit Null ersetzt. Die Funktion EPSX0 ist im ARITHMETIC Menü nicht enthalten, sondern kann nur über die Funktion catalog (N) gestartet werden. Beispiel:

EPSX0(‘X^3-1.2E-12*X^2+1.2E-6*X+6.2E-11)=
‘X^3-0*X^2+.0000012*X+0’.

Mit EVAL: ‘X^3+.0000012*X+0’.

Die Funktion PEVAL

Die Funktion PEVAL (Polynomial EVALuation – Auswertung des Polynoms) kann zur Auswertung eines Polynoms $p(x) = a_n\cdot x^n+a_{n-1}\cdot x^{n-1}+ \ldots + a_2\cdot x^2+a_1\cdot x+ a_0$, verwendet werden, wenn [a_n, a_{n-1}, … a_2, a_1, a_0] ein Array von Koeffizienten und a den Wert von x_0 darstellt. Das Ergebnis ist die Auswertung von $p(x_0)$. Die Funktion PEVAL ist im ARITHMETIC Menü nicht enthalten, sondern kann nur über die Funktion catalog (N) gestartet werden. Beispiel:

PEVAL([1,5,6,1],5) = 281.

Die Funktion TCHEBYCHEFF

Die Funktion TCHEBYCHEFF(*n*) erzeugt das Tchebycheff (oder Chebyshev) Polynom der ersten Art, Grad n, definiert als $T_n(X) = \cos(n\cdot \arccos(X))$. Ist die Integer n negative (n < 0), erzeugt die Funktion TCHEBYCHEFF(*n*) ein Tchebycheff Polynome der zweiten Art, Grad n, definiert als $T_n(X) = \sin(n\cdot \arccos(X))/\sin(\arccos(X))$. Beispiele:

TCHEBYCHEFF(3) = 4*X^3-3*X
TCHEBYCHEFF(-3) = 4*X^2-1

Brüche

Brüche können mit den Funktionen EXPAND und FACTOR, aus dem Menü ALG (,×) erweitert bzw. faktoriert werden. So zum Beispiel:

EXPAND('(1+X)^3/((X-1)(X+3))') = '(X^3+3*X^2+3*X+1)/(X^2+2*X-3)'
EXPAND('(X^2*(X+Y)/(2*X-X^2)^2') = '(X+Y)/(X^2-4*X+4)'
EXPAND('X*(X+Y)/(X^2-1)') = '(X^2+Y*X)/(X^2-1)'
EXPAND('4+2*(X-1)+3/((X-2)*(X+3))-5/X^2') =
'(2*X^5+4*X^4-10*X^3-14*X^2-5*X)/(X^4+X^3-6*X^2)'

FACTOR('(3*X^3-2*X^2)/(X^2-5*X+6)') = 'X^2*(3*X-2)/((X-2)*(X-3))'
FACTOR('(X^3-9*X)/(X^2-5*X+6)') = 'X*(X+3)/(X-2)'
FACTOR('(X^2-1)/(X^3*Y-Y)') = '(X+1)/((X^2+X+1)*Y)'

Die Funktion SIMP2

Die Funktionen SIMP2 und PROPFRAC werden zur Vereinfachung von Brüchen bzw. Erzeugung eines reinen Bruches, verwendet. Die Funktion SIMP2 benötigt als Argument zwei Zahlen oder Polynome, welche den Zähler und Nenner eines rationellen Bruches darstellen und gibt den vereinfachten Zähler und Nenner dieser zurück. So zum Beispiel: SIMP2('X^3-1','X^2-4*X+3') = { 'X^2+X+1','X-3'}.

Die Funktion PROPFRAC

Die Funktion PROPFRAC konvertiert einen rationellen Bruch in einen "reinen" Bruch, d.h. einen Integerwert einem Bruchteil hinzugefügt, falls eine derartige Zerlegung möglich ist. So zum Beispiel:

PROPFRAC('5/4') = '1+1/4'
PROPFRAC('(x^2+1)/x^2') = '1+1/x^2'

Die Funktion PARTFRAC

Die Funktion PARTFRAC zerlegt einen rationellen Bruch in Teilbrüche, die zusammen den ursprünglichen Bruch ergeben. So zum Beispiel:

PARTFRAC('(2*X^6-14*X^5+29*X^4-37*X^3+41*X^2-16*X+5)/(X^5-7*X^4+11*X^3-7*X^2+10*X)') =
'2*X+(1/2/(X-2)+5/(X-5)+1/2/X+X/(X^2+1))'

Diese Technik ist besonders bei der Berechnung von Integralen (siehe Kapitel über Calculus) mit rationellen Brüchen von Nutzen.

Haben Sie den Complex Modus aktiviert, sieht das Ergebnis wie folgt aus:

'2*X+(1/2/(X+i)+1/2/(X-2)+5/(X-5)+1/2/X+1/2/(X-i))'

Die Funktion FCOEF

Die Funktion FCOEF erzeugt einen rationellen Bruch, wenn die Wurzeln und Pole des Bruches bekannt sind.

Anmerkung: Wenn wir einen rationellen Bruch F(X) = N(X)/D(X) haben, können die Wurzeln dieses Bruches mit der Gleichung N(X) = 0 und die Pole über D(X) = 0 berechnet werden.

Die Eingabe für die Funktion bildet einen Vektor von Wurzeln, gefolgt von deren Mehrwertigkeit (d.h. wie oft kann eine Wurzel wiederholt werden) und die Pole gefolgt von deren Mehrwertigkeit als negative Zahl dargestellt. So z.B, wenn wir einen Bruch erstellen wollen, dessen Wurzeln zwei Mehrwertigkeiten 1,0 mit Mehrwertigkeit 3 und -5 mit Mehrwertigkeit 2, enthalten, die Pole 1 mit Mehrwertigkeit 2 und -3 mit Mehrwertigket 5, gehen Sie wie folgt vor:

FCOEF([2 1 0 3 –5 2 1 -2 -3 -5]) = '(X--5)^2*X^3*(X-2)/(X--3)^5*(X-1)^2'

Drücken Sie EVAL erhalten Sie:

'(X^6+8*X^5+5*X^4-50*X^3)/(X^7+13*X^6+61*X^5+105*X^4-45*X^3-297*X^2-81*X+243)'

Die Funktion FROOTS

Die Funktion FROOTS erzeugt die Wurzeln und Pole eines Bruches. Wenn wir z.B. die Funktion FROOTS auf obiges Ergebnis anwenden, erhalten wir: [1 –2 –3 –5 0 3 2 1 –5 2]. Das Ergebnis zeigt Pole gefolgt von deren Mehrwertigkeit als negative Zahl und Wurzeln gefolgt von deren Mehrwertigkeit in Form einer positiven Zahl. In diesem Fall sind die Pole (1, -3) mit entsprechender Mehrwertigkeit (2,5), und die Wurzeln (0, 2, -5) mit der entsprechenden Mehrwertigkeit (3, 1, 2).

Ein weiteres Beispiel ist: FROOT('(X^2-5*X+6)/(X^5-X^2)') = [0,–2,1, –1,3,1,2, 1], d.h., Pole = 0 (2), 1(1) und Wurzeln = 3(1), 2(1). Befinden Sie sich im Complex Modus, würde ihr Ergebnis so aussehen: [0 –2 1 –1 '-((1+i*√3)/2' –1].

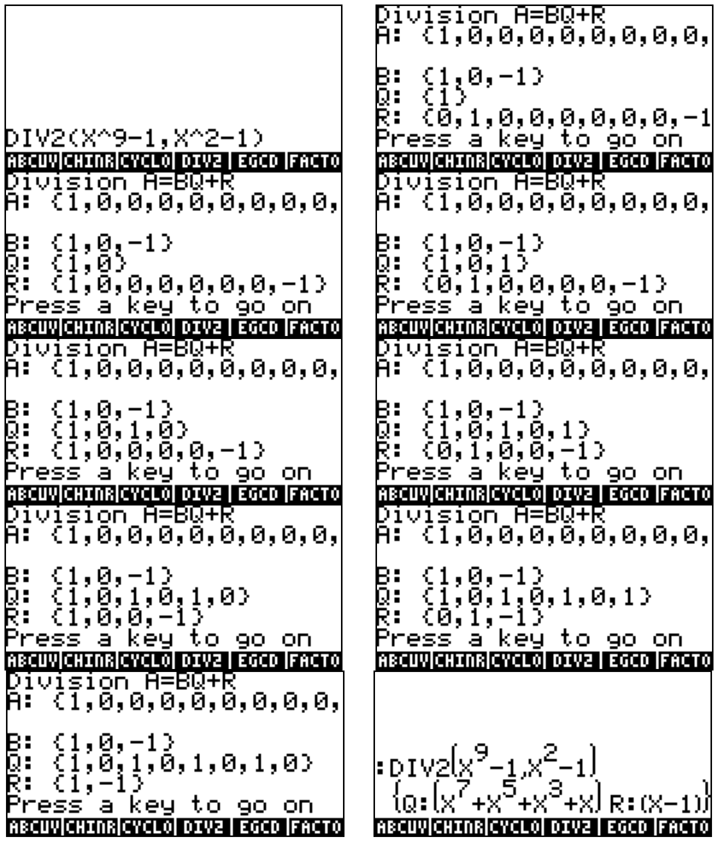
Step-by-Step Operationen mit Polynomen und Brüchen

Stellen Sie das CAS auf Step/step, wird der Rechner schrittweise Vereinfachungen von Brüchen und Operationen mit Polynomen anzeigen. Dies ist besonders bei der synthetischen Division nützlich, um die einzelnen Schritte der Division zu sehen. Das Beispiel der Division

$$\frac{X^3 - 5X^2 + 3X - 2}{X - 2}$$

wird ausführlich in Anhang C gezeigt. In nachfolgendem Beispiel wird eine längere synthetische Division angezeigt:

$$\frac{X^9 - 1}{X^2 - 1}$$



Das Menü CONVERT und algebraische Operationen

Das Menü CONVERT wird über die Tasten ← CONVERT (die Taste 6) gestartet. Das Menü zeigt alle Umwandlungs-Menüs im Rechner an. Die Liste dieser Menüs folgt als Nächstes:

Die in jedem der Untermenüs vorhandenen Funktionen werden nachfolgend gezeigt.

Menü Konvertierung von UNITS (Einheiten) (Option 1)

Dieses Menü entspricht dem Menü UNITS unter Verwendung von ↱ UNITS . Die Anwendungen dieses Menüs werden ausführlich in Kapitel 3 erläutert.

Konvertierungs-Menü BASE (Option 2)

Dieses Menü entspricht dem Menü BASE unter Verwendung von ↱ BASE . Die Anwendungen dieses Menüs werden ausführlich in Kapitel 19 erläutert.

Konvertierungs-Menü TRIGONOMETRIC (Option 3)

Dieses Menü entspricht dem Menü TRIG unter Verwendung von ↱ TRIG . Die Anwendungen dieses Menüs werden ausführlich in diesem Kapitel erläutert.

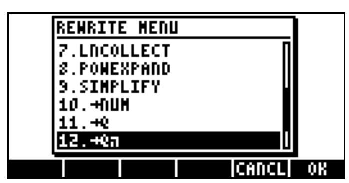
Konvertierungs-Menü MATRIZEN (Option 5)

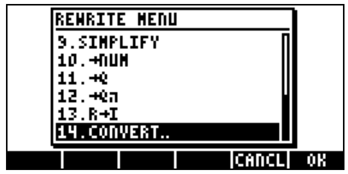
Dieses Menü enthält zusätzlich die nachfolgenden Funktionen:

Diese Funktionen werden ausführlich in Kapitel 10 erläutert.

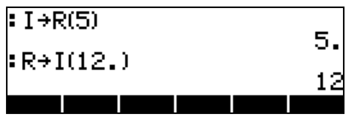
Konvertierungs-Menü REWRITE (Option 4)

Dieses Menü enthält zusätzlich die nachfolgenden Funktionen:

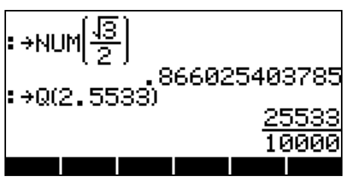


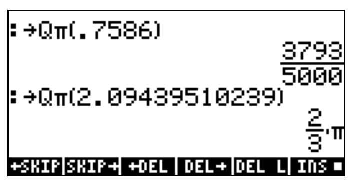


Die Funktionen I→R und R→I werden dazu verwendet aus einer Integer (I) eine reelle (R) oder umgekehrt zu konvertieren. Integerzahlen werden ohne Dezimalpunkte angegeben, während reelle Zahlen Integerwerte darstellen und einen Dezimalpunkt am Ende besitzen, z.B.



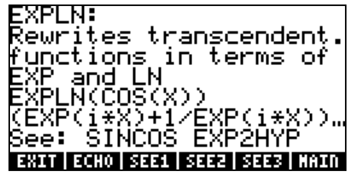
Die Funktion →NUM hat den gleichen Effekt die die Tastenkombination →NUM (der Taste ENTER zugeordnet). Die Funktion →NUM konvertiert ein symbolisches Ergebnis in ein Gleitpunkt-Ergebnis. Die Funktion →Q konvertiert ein Gleitpunktwert in einen Bruch. Die Funktion →Qπ konvertiert einen Gleitpunktwert in einen Bruch von π, wenn ein Bruch von π, für die Zahl gefunden werden kann; andernfalls konvertiert er diese Zahl in einen Bruch. Nachfolgend einige Anwendungsbeispiele dieser Funktionen.



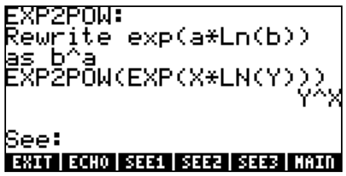


Aus den Funktionen des Menüs REWRITE, stehen die Funktionen DISTRIB, EXPLN, EXP2POW, FDISTRIB, LIN, LNCOLLECT, POWEREXPAND und SIMPLIFY für algebraische Ausdrücke zur Verfügung. Viele dieser Funktionen werden in diesem Kapitel vorgestellt. Der Vollständigkeit halber, zeigen wir die Einträge in der Hilfefunktion für diese Funktionen.

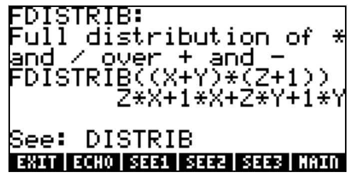
DISTRIB EXPLN



EXP2POW

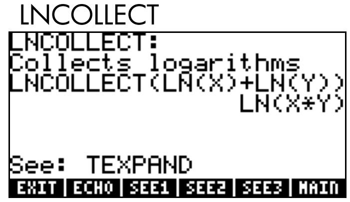


FDISTRIB



LIN

LNCOLLECT



POWEREXPAND

SIMPLIFY

Kapitel 6
Lösung für Einzelgleichungen

In diesem Kapitel behandeln wir die Funktionen des Rechners zur Lösung von Einzelgleichungen der Form f(X) = 0. Der Taste 7 sind zwei Gleichungs-Lösungs-Menüs zugeordnet, der symbolische SOLVer (Löser) (← S.SLV) und der NUMerische SOLVer (Löser) (→ NUM.SLV). Nachfolgend werden einige Funktionen aus diesen Menüs beschrieben. Ändern Sie für diese Beispiele den CAS-Modus auf Complex (siehe Kapitel 2).

Symbolische Lösung algebraischer Gleichungen

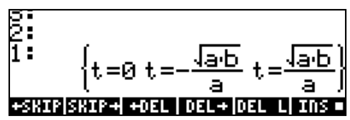
Nachfolgend werden einige Funktionen aus dem Menü symbolischer Löser beschrieben. Aktivieren Sie das Menü über die Tastenkombination. Mit dem System Flag 117 auf *CHOOSE* Kästchen gesetzt, werden folgende Menü-Einträge aufgelistet:

Die Funktionen DESOLVE und LDEC werden zur Lösung von Differentialgleichungen, Thema eines anderen Kapitels, verwendet und deshalb in diesem nicht erläutert. Ähnlich die Funktion LINSOLVE, die zur Lösung von mehrfachen linearen Gleichungen dient und in einem anderen Kapitel präsentiert wird. Die Funktionen ISOL und SOLVE können zur Lösung der Unbekannten in einer Polynom-Gleichung verwendet werden. Die Funktion SOLVEX löst eine Polynomgleichung, in welcher die Standard CAS Variable VX (standardmäßig 'X') die Unbekannte ist. Schließlich die Funktion ZEROS, welche Nullen, Wurzeln oder ein Polynom bereitstellt. Für alle Funktionen, außer ISOL, sind Einträge im S.SLV Menü über die CAS Hilfefunktion (TOOL NXT HELP) vorhanden.

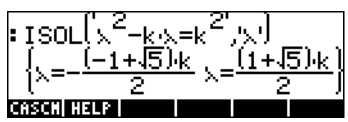
Funktion ISOL

Mit der Funktion ISOL (Gleichung, Variable) erhalten Sie dieLösung(en) für *Gleichung* durch Isolierung der *Variablen*. So z.B. um *t* in der Gleichung at^3-bt = 0, mit dem Rechner im ALG-Modus, zu ermitteln, können wir wie folgt vorgehen:

Im RPN-Modus erhalten wir das gleiche Ergebnis, wenn wir die Gleichung in den Stack, gefolgt von der Variablen schreiben und anschließend die Funktion ISOL eingeben. Bevor Sie die Funktion ISOL ausführen, sollte die Anzeige im RPN-Modus, wie in der Abbildung auf der linken Seite, aussehen. Nachdem Sie die Funktion ISOL ausgeführt haben, sieht ihre Anzeige, wie in der rechten Abbildung aus:

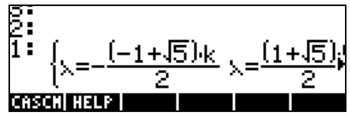


Das erste Argument in ISOL kann ein Ausdruck – wie oben aufgeführt – oder eine Gleichung sein. Versuchen Sie z.B. im ALG-Modus:



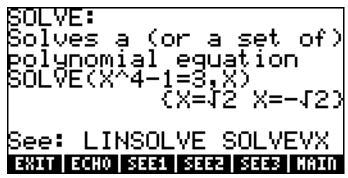
Anmerkung: Um das Istgleichzeichen (=) in einer Gleichung zu schreiben, verwenden Sie die Tastenfolge ↱ = (der Taste +/- zugeordnet).

Das gleiche Problem kann im RPN-Modus, wie unten angezeigt gelöst werden (Abbildungen zeigen den RPN-Stack vor und nachdem die Funktion ISOL angewendet wurde):

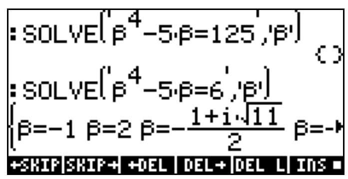
 

Funktion SOLVE

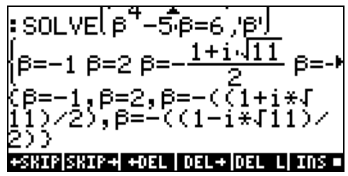
Die Funktion SOLVE hat die gleiche Syntax wie die Funktion ISOL, nur dass SOLVE auch zur Lösung eines Sets von Polynom-Gleichungen verwendet werden kann. Untenstehend, der Hilfetext für die Funktion SOLVE, mit der Lösung der Gleichung X^4 – 1 = 3:



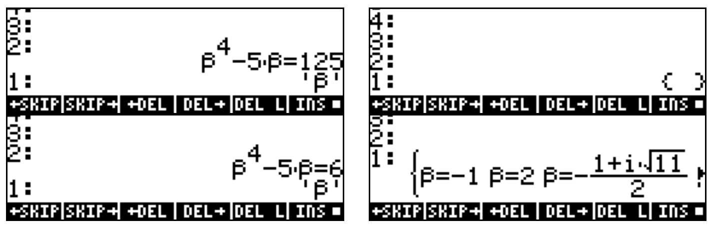
Nachfolgende Beispiele zeigen die Funktion SOLVE im ALG- und im RPN-Modus:



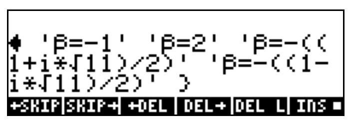
Die obige Abbildung zeigt zwei Lösungen. In der ersten, β^4-5β =125 findet SOLVE keine Lösungen { }. In der zweiten Abbildung hingegen, β^4 - 5β = 6, findet SOLVE gleich vier Lösungen, welche in der letzten Ausgabezeile angezeigt sind.   Die letzte Lösung ist nicht sichtbar, weil die Anzahl der Buchstaben der Lösung größer als die Breite der Anzeige des Displays ist. Sie können aber alle Lösungen, mit Hilfe der Pfeiltaste (▼), welche von einer Zeile des Zeileneditors in die andere umschaltet, ansehen (dieser Vorgang kann jederzeit benutzt werden, wenn die Ausgabezeile länger als die Breite der Rechner-Displays ist):



Die entsprechende Anzeige im RPN-Modus für diese beiden Beispiele, vor und nach der Anwendung der Funktion SOLVE, ist nachstehend zu sehen:

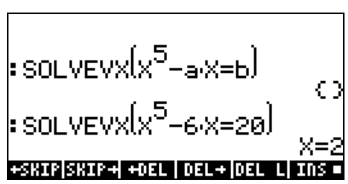


Benutzen Sie die Pfeiltaste ▽ in diesem Modus wird der Zeileneditor gestartet:



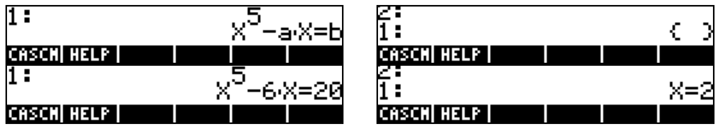
Funktion SOLVEVX

Die Funktion SOLVEVX löst eine Gleichung für die Standard CAS-Variable in der reservierten Variablen VX. Standarmäßig ist der Wert dieser Variablen 'X'. Nachfolgende Beispiele, im ALG-Modus mit VX = 'X':

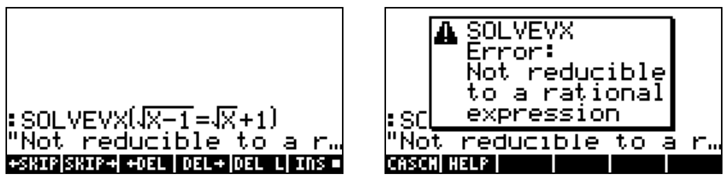


Im ersten Fall konnte SOLVEVX keine Lösung finden. Im zweiten Fall, hat SOLVEVX eine einzige Lösung gefunden, X = 2.

Nachfolgend die Anzeige der beiden Beispiele im RPN Stack (vor und nach Anwendung der Funktion SOLVEVX):

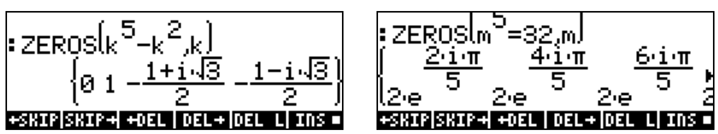


Wird die Gleichung als Argument für die Funktion SOLVEVX benutzt, muss diese auf einen rationalen Ausdruck vereinfacht werden können. So z.B. wird die nachfolgende Gleichung von SOLVEVX nicht verarbeitet:

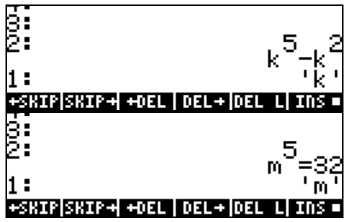


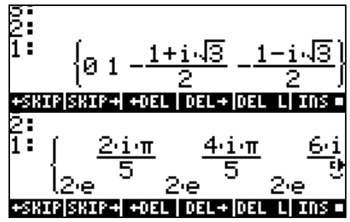
Funktion ZEROS

Die Funktion ZEROS errechnet Lösungen einer Polynomgleichung, ohne deren Mehrwertigkeit anzuzeigen. Als Eingabe für die Funktion wird der Ausdruck für die Gleichung und der Name der Variablen die zu lösen ist benötigt. Nachfolenge Beispiele im ALG-Modus:



Um die Funktion ZEROS im RPN-Modus zu verwenden, muss zuerst der Polynomausdruck eingegeben werden, dann die zu lösende Variable, anschließend dann die Funktion ZEROS. Die nachfolgenden Abbildungen zeigen den RPN-Stack vor und nach Anwendung der Funktion ZEROS auf die beiden obigen Beispiele:

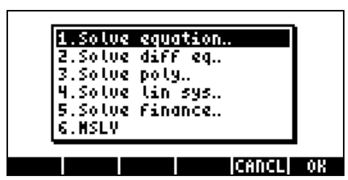




Die Funktionen des oben aufgeführten symbolischen Lösers ermitteln Lösungen für rationale Gleichungen (hauptsächlich Polynom-Gleichungen). Wenn alle Koeffizienten der zu lösenden Gleichung numerisch sind, ist auch eine numerische Lösung über den numerischen Löser des Rechners möglich.

Menü numerischer Löser

Der Rechner bietet eine starke Umgebung zur Lösung von einzelnen algebraischen oder transzendenten Gleichungen. Um sich zu dieser Umgebung Zugang zu verschaffen, starten sie den numerischen Löser (NUM.SLV) mit Hilfe von ↱ NUM.SLV . Dieses erstellt ein Drop-Down Menü, mit folgenden Optionen:



Die Position *2. Solve diff eq..* wird in einem späteren Kapitel, unter Differentialgleichungen näher behandelt. Position *4. Solve lin sys..* wird in einem späteren Kapitel über Matrizen behandelt. Position *6. MSLV* (Mehrfacher SoLVer (Gleichungslöser)) wird im nächsten Kapitel erläutert. Nachfolgend präsentieren wir Anwendungen zu den Positionen *3. Solve poly..*, *5. Solve finance*, und *1. Solve equation..*, (in dieser Reihenfolge). In Anhang 1-A am Ende von Kapitel 1, finden Sie Anleitungen zur Benutzung von Eingabeformularen und Beispiele für Anwendungen mit dem numerischen Löser.

Anmerkungen:

1. Wann immer Sie eine Lösung in der NUM.SLV Anwendung berechnen, wird der gefundene Wert in den Stack geschrieben. Dies erweist sich als nützlich, wenn sie diesen Wert für spätere Operationen benötigen.
2. Jedes Mal, wenn Sie eine Anwendung im NUM.SLV Menü starten, wird eine oder mehrere Variablen erzeugt.

Polynomgleichungen

Wenn Sie die Option *Solve poly…* in der SOLVE Umgebung Ihres Rechners benutzen, können Sie:

(1) Lösungen zu einer Polynomgleichung finden;
(2) die Koeffizienten des Polynoms, mit einer bekannten Anzahl von Wurzeln ermitteln;
(3) einen algebraischen Ausdruck für das Polynom als Funktion von X ermitteln.

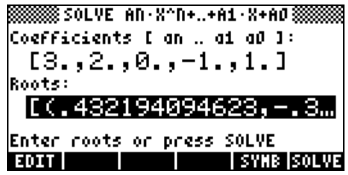
Lösungen zu einer Polynomgleichung berechnen

Eine Polynomgleichung ist eine Gleichung die wie folgt aussieht: $a_nx^n + a_{n-1}x^{n-1} + \ldots + a_1x + a_0 = 0$. Der fundamentale Lehrsatz der Algebra weist darauf hin, dass es *n* Lösungen zu jeder Polynom-Gleichung *n*-ten Grades gibt. Nichts desto trotz können einige Lösungen aber auch komplexe Zahlen sein. Als Beispiel lösen Sie die Gleichung: $3s^4 + 2s^3 - s + 1 = 0$.

Wir möchten die Koeffizienten der Gleichung in einen Vektor $[a_n, a_{n-1}, a_1\ a_0]$ setzen. Für dieses Beispiel benutzen wir den Vektor [3,2,0,-1,1]. Um diese Polynomgleichung mit dem Rechner zu lösen, versuchen Sie folgendes:

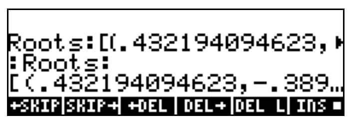
↱ NUM.SLV ▼ ▼ OK	Wählen Sie Solve poly…
↰ [] 3 ↱ , 2 ↱ , 0 ↱ , 1 +/- ↱ , 1 OK	Tragen Sie die Koeffizienten in einen Vektor ein
SOLVE	sen der Gleichung

In der Anzeige wird die Lösung wie folgt aussehen:



Drücken Sie `ENTER` um zum Stack zurückzukehren. Der Stack zeigt die folgenden Ergebnisse im ALG-Modus an (die gleichen Ergebnisse werden auch im RPN-Modus angezeigt):

Um alle Lösungen anzuzeigen, drücken Sie die Pfeiltaste ▼ zur Navigation im Zeileneditor:



Alle Lösungen sind komplexe Zahlen: (0.432,-0.389), (0.432,0.389), (-0.766, 0.632), (-0.766, -0.632).

Anmerkung: Beachten Sie, dass komplexe Zahlen im Rechner als geordnete Paare dargestellt werden, wobei die erste Zahl im Paar den reellen Teil und die zweite den imaginären Teil darstellt. So z.B. wird die Zahl (0,432,-0,389), eine komplexe Zahl, normalerweise als 0,432 – 0,389*i* dargestellt, wobei *i* den imaginären Teil, d.h. $i^2 = -1$ darstellt.

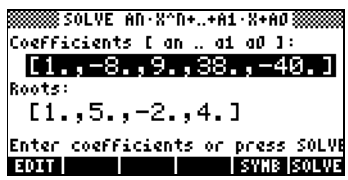
Anmerkung: Der fundamentale Lehrsatz der Algebra weist darauf hin, dass es *n* Lösungen zu jeder Polynom-Gleichung *n*-ten Grades gibt. Es gibt einen weiteren Lehrsatz in der Algebra, welcher darauf hinweist, dass, wenn eine Lösung einer Polynom-Gleichung mit reellen Koeffizienten eine komplexe Zahl ist, dann ist die konjugierte dieser Zahl auch eine Lösung. Mit anderen Worten, komplexe Lösungen zu einer Polynom-Gleichung mit reellen Koeffizienten kommen paarweise. Das bedeutet, dass Polynomgleichungen mit reellen Koeffizienten von ungeraden Zahlen, haben mindestens eine reelle Lösung.

Erzeugen von Polynom-Koeffizienten , wenn die Wurzeln des Polynoms bekannt sind

Angenommen Sie wollen ein Polynom erstellen dessen Wurzeln die Zahlen [1, 5, -2, 4] sind. Um den Rechner für diesen Zweck zu nutzen, führen Sie folgende Schritte aus:

NUM.SLV OK	Wählen Sie Solve poly…
[] 1 , 5	
, 2 +/- , 4 OK	Tragen Sie die Wurzeln in einen Vektor ein
SOLVE	Solve (Lösung) für

KoeffizientenDrücken Sie ENTER, um zum Stack zurückzukehren, die Koeffizienten werden im Stack angezeigt.



Drücken Sie ▼, um alle Koeffizienten im Zeileneditor anzuzeigen.

Anmerkung: Möchten Sie ein Polynom mit reellen Koeffizienten erhalten, dies aber komplexe Wurzeln besitzt, müssen Sie die komplexen Wurzeln als paare von konjugierten Zahlen eingeben. Um dies zu veranschaulichen, erstellen wir ein Polynom mit den Wurzeln [1 (1,2) (1,-2)]. Überprüfen Sie, dass das erhaltene Polynom nur reelle Koeffizienten enthält. Versuchen Sie auch ein Polynom mit den Wurzeln [1 (1,2) (-1,2)] zu erstellen und überprüfen Sie, dass das daraus resultierende Polynom komplexe Koeffizienten enthält.

Erstellen eines algebraischen Ausdrucks für das Polynom

Sie können bei der Erstellung eines algebraischen Ausdrucks für ein Polynom, mit vorgegebenen Koeffizienten der Wurzeln des Polynoms, den Rechner benutzen. Der ermittelte Ausdruck wird als Standard CAS Variable X ausgegeben. (Die nachfolgenden Beispiele zeigen wie Sie X mit einer anderen Variable über die Funktion |).

Um den algebraischen Ausdruck mit Hilfe der Koeffizienten zu erstellen, nehmen Sie nachfolgendes Beispiel. Nehmen wir an die Koeffizienten des Polynoms sind [1,5,-2,4]. Verwenden Sie dazu folgende Tastenfolge:

‍→ NUM.SLV ▼ ▼ OK	Wählen Sie Solve poly…
← [] 1 → , 5	Tragen Sie die Koeffizienten in einen Vektor ein
→ , 2 +/- → , 4 OK	
▲ SYMB	Erzeugen Sie den symbolischen Ausdruck
ENTER	Zurück zum Stack.

Der so ermittelte Ausdruck wird im Stack wie folgt angezeigt:

'X^3+5*X^2+-2*X+4'.

Um den algebraischen Ausdruck mit Hilfe der Wurzeln zu erstellen, nehmen Sie nachfolgendes Beispiel. Nehmen wir an die Wurzeln des Polynoms sind [1,3,-2,1]. Verwenden Sie dazu folgende Tastenfolge:

→ NUM.SLV ▼ ▼ OK	Wählen Sie Solve poly…
▼ ← [] 1 → , 3	Tragen Sie die Wurzeln in einen Vektor ein
→ , 2 +/- → , 1 OK	
▼ SYMB	Erzeugen Sie den symbolischen Ausdruck
ENTER	Zurück zum Stack.

Der so generierte Ausdruck wird im Stack wie folgt angezeigt:

'(X-1)*(X-3)*(X+2)*(X-1)'.

Um die Ergebnisse zu erweitern, können Sie den Befehl EXPAND verwenden. Der daraus resultierende Ausdruck sieht wie folgt aus:

'X^4+-3*X^3+ -3*X^2+11*X-6'.

Ein weiterer Ansatz einen Ausdruck für das Polynom zu bekommen, ist erst die Koeffizienten zu generieren, dann den algebraischen Ausdruck mit hervorgehobenen Koeffizienten. Versuchen Sie in diesem Fall:

NUM.SLV ▼ ▼ OK	Wählen Sie Solve poly…
▼ [] 1 , 3	Tragen Sie die Wurzeln in einen Vektor ein
, 2 +/- , 1 OK	
SOLVE	Solve (Lösung) für Koeffizienten
▼ SYMB	Erzeugen Sie den symbolischen Ausdruck
ENTER	Zurück zum Stack.

Der so ermittelte Ausdruck wird im Stack wie folgt angezeigt: 'X^4+-3*X^3+ -3*X^2+11*X+-6*X^0'. Die Koeffizienten werden in Stack Ebene 2 angezeigt.

Finanzmathematische Berechnungen

Die Berechnungen in Position *5. Solve finance..* im numerischen Löser (*NUM.SLV*) werden zur Berechnung des Zeitwertes von Geld wichtig in der technischen Wirtschaft   und anderer Finanz-Anwendungen, verwendet. Diese Anwendung kann mit der Tastenkombination $ FINANCE (Taste 9 zugeordnet) gestartet werden. Bevor wir diese Lösungsumgebung im Detail schildern, präsentieren wir einige Definitionen, welche zum Verständnis finanzmathematischer Operationen im Rechner notwendig sind.

Definitionen

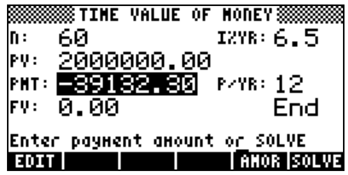
Es kommt häufig vor, dass man zur Entwicklung eines Projektes, Geld von einem Finanzinstitut oder aus öffentlichen Mitteln borgen muss.   Die geborgte Geldsumme wird als *aktueller Wert* (PV) bezeichnet. Dieser Geldbetrag muss in *n* Raten (Zeitabschnitten) (normalerweise Vielfache oder Untereinheiten eines Monats), unterliegen einer jährlichen *Zinsrate* von I%YR. , zurückgezahlt werden. Die *Anzahl von Zeitabschnitten pro Jahr* (P/YR) ist eine Integerzahl von Zeitabschnitten, in welches das Jahr, zum Zwecke der Zurückzahlung des Kredits, unterteilt wurde. Standardwerte von P/YR sind 12 (eine Zahlung pro Monat), 24 (zwei Zahlungen pro Monat) oder 52 (wöchentliche Zahlungen). Die (*Zahlung) payment*(PMT) ist die Summe, die der Leiher an den Verleiher am Anfang oder am Ende jedes Zeitabschnittes *n* der Leihfrist bezahlen muss. Der *zukünftige Wert* des Geldes (FV) ist der Wert der ausgeliehenen Geldsumme am Ende von *n* Zeitschabschnitten. Normalerweise erfolgt die Bezahlung jeweils am Ende eines Zeitabschnittes, so dass der Leiher am End des ersten Zeitabschnittes mit der Bezahlung beginnt und die gleiche feste Summe am Ende des zweiten, dritten usw. Zeitabschnittes bis hin zum letzten Zeitabschnitt *n* bezahlt.

Beispiel 1 – Berechnung der Bezahlung für einen Kredit

Wenn 2 Millionen Dollar, bei einem jährlichen Zinssatz von 6,5% mit einer Leihfrist von 60 Monaten ausgeliehen werden, wie viel beträgt die monatliche Ratenzahlung? Um die Schulden komplett innerhalb von 60 Monaten zu bezahlen, sollten die zukünftigen Werte des Darlehens Null betragen. Um dies über die finanzmathematischen Merkmale des Rechners durchzuführen, benutzen wir die nachfolgenden Werte: n = 60, I%YR = 6,5, PV = 2000000, FV = 0, P/YR = 12. Um die Daten einzugeben und die Zahlung, PMT, zu berechnen, gehen Sie wie folgt vor:

← FINANCE	Starten Sie die Eingabemaske für Finanzmathematik
60 OK	Geben Sie n = 60 ein
6,5 OK	Geben Sie I%YR = 6,5 % ein
2000000 OK	Geben Sie PV = 2.000.000 US$ ein
▼	Übergehen Sie PMT, da wir diese berechnen wollen
0 OK	Geben Sie FV = 0 ein, die Option End wird hervorgehoben
▲ ◄ SOLVE	Heben Sie PMT hervor, um diesen Wert zu ermitteln

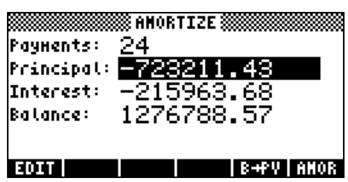
Die Anzeige sieht wie folgt aus:



In der Anzeige erscheint der Wert für PMT als –39.132,30, d.h. der Kreditnehmer wird eine monatliche Rate von US $ 39.132,30 am Ende jedes Monats während der kommenden 60 Monate zahlen, um den Gesamtbetrag zurückzuzahlen. Der Grund warum der Wert PMT negativ ausgefallen ist, besteht darin, dass der Rechner die Werte aus der Sicht des Kreditnehmers betrachtet. Der Kreditnehmer besitzt ein Plus von US $ 2.000.000,00 in der Zeitspanne t = 0, dann beginnt er mit der Zahlung, so dass jedes Mal – US $ 39132,30 in den Perioden t = 1, 2, …, 60 hinzuaddiert werden. Bei t = 60, liegt der tatsächliche Nettowert des Entleihers bei Null. Wenn Sie nun den Betrag von US $ 39.132,30 nehmen und diesen mit 60 Zahlungen multiplizieren, wird die tatsächlich zurückgezahlte Gesamtsumme US $ 2.347.937,79 betragen. Somit macht der Verleiher einen Nettogewinn von $ 347.937,79 in den 5 Jahren in welchen er das Projekt des Kreditnehmers finanziert.

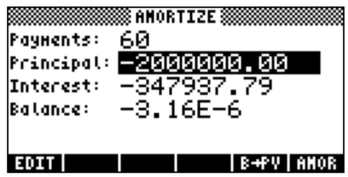
Beispiel 2 – Berechnung der Amortisation für einen Kredit

Sie erhalten die gleiche Lösung zu dem Problem aus Beispiel 1, wenn Sie die Taste AMOR , welche für AMORTIZATION steht, drücken. Diese Option wird dazu benutzt, um zu sehen wieviel am Ende einer bestimmten Leihfrist insgesamt bezahlt (amortisiert) wurde. Nehmen wir an, das wir 24 Leihfristen in der ersten Zeile der Amortisationsanzeige verwenden, d.h. 2 4 OK. Drücken Sie anschließend AMOR. Sie erhalten folgendes Ergebnis:



Diese Anzeige wird so interpretiert, dass nach 24 Monaten Schuldenrückzahlungen der Kreditnehmer insgesamt US $ 723.211,43 in seinen Gesamtschuldenbetrag eingezahlt und US $ 215.963,68 an Zinsen bezahlt hat. Der Kreditnehmer muss noch eine Differenz von US $ 1.276.788,57 innerhalb der nächsten 36 Monate zurückzahlen.

Überprüfen Sie was passiert, wenn Sie 60 bei den Zahlungen eingeben: starten Sie die Amortisationsanzeige drücken Sie dann OK AMOR. Die Anzeige sieht nun wie folgt aus:



Das bedeutet, dass am Ende von 60 Monaten der entliehene Betrag von US $ 2.000.000,000 zusammen mit den Zinsen von US $ 347.937,79 abbezahlt wurde, der Differenzbetrag aber noch US $ 0,000316 beträgt, welche der Kreditnehmer dem Verleiher schuldet, ist. Sicherlich sollte der Differenzbetrag aber Null sein. Der im Display angezeigte Wert, ist ein Rundungsfehler aus der numerischen Lösung.

Drücken Sie ON oder ENTER zweimal, um zur Normalanzeige des Rechners zurückzukehren.

Beispiel 3 – Berechen der Zahlung am Anfang der Zeitspanne
Lösen wir das gleiche Problem wie in den Beispielen 1 und 2, aber mit der Option, dass die Zahlung am Anfang des gleichen Abschnitts stattfindet. Verwenden Sie:

← FINANCE	Starten Sie die Eingabemaske für Finanzmathematik
60 OK	Geben Sie n = 60 ein
6,5 OK	Geben Sie I%YR = 6,5 % ein
2000000 OK	Geben Sie PV = 2.000.000 US$ ein
▼	Übergehen Sie PMT, da wir diese berechnen wollen
0 OK	Geben Sie FV = 0 ein, die Option End wird hervorgehoben

CHOOS ▲ OK	Ändern Sie die Option für die Zahlung auf *Begin (Anfang)*
▲ ◄ SOLVE	Heben Sie PMT hervor, um diesen Wert zu lösen

In der Anzeige erscheint der Wert für PMT als -38.921,47, d.h. der Kreditnehmer wird eine monatliche Rate von US $ 38.921,48 am *Anfang* jedes Monats während der kommenden 60 Monate zahlen, bis er den Gesamtbetrag zurückgezahlt hat. Beachten Sie, dass der Betrag den der Kreditnehmer monatlich zu bezahlen hat, wenn er diesen am Anfang jeden Monats bezahlt, geringfügig niedriger als am Ende des gleichen Monats ist. Der Grund dafür ist, dass der Verleiher Zinsguthaben für die Bezahlungen am Anfang des Monats bekommt, und somit die Schuldlast des Kreditnehmers etwas verringert.

Anmerkungen:

1. Die finanzmathematische Umgebung erlaubt es jeden x-beliebigen darin enthaltenen Wert, d.h. n, I%YR, PV, FV, P/Y, wenn die anderen Werte der Ausleihe bekannt sind, zu berechnen. Heben Sie einfach den Wert, den Sie berechnen möchten hervor und drücken SOLVE. Das Ergebnis wird im hervorgehobenen Feld angezeigt.

2. Die in der finanzmathematischen Umgebung des Rechners berechneten Werte, werden mit deren entsprechenden Kennung (Kennzeichen zur Identifizierung) in den Stack kopiert.

Löschen von Variablen

Wenn Sie die finanzmathematische Umgebung Ihres Rechners zum ersten Mal nutzen, werden im HOME Verzeichnis oder in irgendeinem Unterverzeichnis die Variablen n I%YR PV PMT PYR FV erzeugt, um die entsprechenden Faktoren, für die verschiedenen Berechnungen, zu speichern. Sie können sich den Inhalt dieser Variablen wie folgt anzeigen lassen:

↱ n ↱ I%YR ↱ PV ↱ PMT ↱ PYR ↱ FV.

Sie können diese für zukünftige Anwendungen im Verzeichnis belassen, oder mit Hilfe der Funktion PURGE aus diesem löschen. Um alle Variablen auf einmal zu löschen, wenn im ALG-Modus, versuchen Sie folgendes:

TOOL PURGE VAR ← {}	Geben Sie PURGE ein und erstellen Sie eine Liste der Variablen
' → n	Geben Sie den Namen der Variablen N ein
▶ → ,	Geben Sie ein Komma ein
' → I%YR	Geben Sie den Namen der Variablen I%YR ein
▶ → ,	Geben Sie ein Komma ein
' → PV	Geben Sie den Namen der Variablen PV ein
▶ → ,	Geben Sie ein Komma ein
' → PMT	Geben Sie den Namen der Variablen PMT ein
▶ → ,	Geben Sie ein Komma ein
' → PYR	Geben Sie den Namen der Variablen PYR ein
▶ → ,	Geben Sie ein Komma ein
' → FV.	Geben Sie den Namen der Variablen FV ein
ENTER	Führen Sie den PURGE Befehl aus

In den nachfolgenden Abbildungen sehen Sie den PURGE Befehl zum löschen aller Variablen im Verzeichnis und das Ergebnis, nachdem Sie den Befehl ausgeführt haben.

Denselben Wert errechnen Sie im RPN-Modus wie folgt:

VAR ← {}	Erstellen Sie eine Liste von Variablen die gelöscht werden sollen
n	Geben Sie den Namen der Variablen N ein
I%YR	Geben Sie den Namen der Variablen I%YR ein

[PV]	Geben Sie den Namen der Variablen PV ein
[PMT]	Geben Sie den Namen der Variablen PMT ein
[PYR]	Geben Sie den Namen der Variablen PYR ein
[FV]	Geben Sie den Namen der Variablen FV ein
[ENTER]	Geben Sie die Liste der Variablen in den Stack
[TOOL] [PURGE]	Löschen Sie die Variablen in der Liste

Bevor Sie den PURGE Befehl eingeben, sieht der RPN-Stack wie folgt aus:

Lösen von Gleichungen mit einer Unbekannten über NUM.SLV

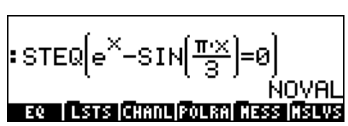
Das Menü NUM.SLV des Rechners bietet in Position *1. Solve equation..* die Lösung verschiedener Typen von Gleichungen in einer einzigen Variablen, einschließlich nicht-linearer algebraischer und transzendenter Gleichungen. Als Beispiel lösen wir nachfolgende Gleichung: e^x-$sin(\pi x/3) = 0$.

Geben Sie den Ausdruck als algebraisches Objekt ein und speichern Sie dies in der Variablen EQ. Die dazu notwendigen Tastenfolgen im ALG-Modus sind:

['] [←] e^x [ALPHA] [←] [X] [▶] [−] [SIN] [←] π
[×] [ALPHA] [←] [X] [÷] [3] [▶] [→] = [0] [▶]
[STO▶] [ALPHA] [E] [ALPHA] [Q] [ENTER]

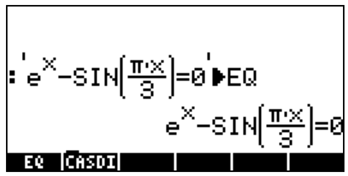
Funktion STEQ

Die Funktion STEQ, die über den Befehl Katalog, [→] CAT , gestartet wird, speichert ein Argument in einer Variablen EQ, z.B. im ALG-Modus:

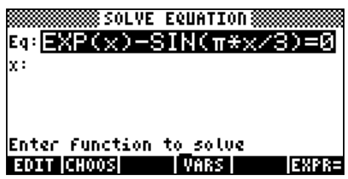


Im RPN-Modus tragen Sie die Gleichung zwischen zwei Apostrophe ein und starten Sie den Befehl STEQ. Somit kann die Funktion STEQ als Kürzel zur Speicherung des Ausdrucks in der Variablen EQ verwendet werden.

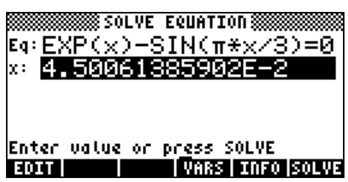
Drücken Sie VAR um die neu erstellte Variable EQ anzuzeigen:



Wechseln Sie anschließend in die SOLVE Umgebung und wählen Sie *Solve equation…*, unter Verwendung von:
→ NUM.SLV OK. Die entsprechende Anzeige sieht wie folgt aus:



Die Gleichung, die wir soeben in der Variablen EQ gespeichert haben, ist bereits ins Feld *Eq* in der Eingabemaske SOLVE EQUATION geladen. Auch ein mit x beschriftetes Feld wird bereit gestellt. Um die Gleichung zu lösen, müssen Sie einfach nur noch das Feld vor dem X markieren: indem Sie die Pfeiltaste ▼ benutzen und dann SOLVE drücken. Die gezeigte Lösung ist X: 4,5006E-2:



Dies ist aber nicht die einzig mögliche Lösung für diese Gleichung. Um z.B. eine negative Lösung zu erhalten, tragen Sie, bevor Sie die Gleichung lösen, eine negative Zahl in das Feld X: ein. Versuchen Sie 3 +/- OK ▼ SOLVE. Die Lösung lautet nun: -3,045.

Lösungsschema für die *Gleichung Solve…*

Der numerische Löser für Gleichungen mit einer Unbekannten funktioniert wie folgt:

- Er erlaubt dem Anwender die Gleichung einzugeben oder den Typ der zu lösenden Gleichung zu wählen CHOOS.
- Er erzeugt eine Eingabemaske mit Eingabefelder entsprechend aller in der Gleichung vorkommenden Variablen, welche in der Variablen EQ gespeichert sind.
- Der Anwender muss die Werte aller vorkommenden Variablen eingeben außer einer.
- Anschließend markiert der Anwender das Feld mit der Unbekannten, für welche die Gleichung gelöst werden soll und drückt dann SOLVE
- Der Anwender kann eine Lösung erzwingen, indem er eine Schätzung der Lösung im entsprechenden Eingabefeld, bevor er die Gleichung löst, vorgibt.

Der Rechner benutzt einen Suchalgorithmus, um ein Interval festzulegen, für welches die Funktion das Vorzeichen ändert, was andeutet, dass es eine Wurzel oder Lösung für die Gleichung gibt. Er verwendet anschließend eine numerische Methode in eine Lösung zu konvergieren.

Die Lösung die der Rechner sucht, wird durch den vorhandenen Anfangswert im Feld der Unbekannten, bestimmt. Ist kein Wert vorhanden, benutzt der Rechner den Standardwert Null.Somit, können Sie mehr als nur eine Lösung zu einer Gleichung suchen, indem Sie diesen Anfangswert im Feld der Unbekannten ändern. Beispiele zu Lösungen für die Gleichung werden nachfolgend gezeigt.

Beispiel 1 – Hooke's Gesetz für Dehnung und Spannung

Die zu verwendende Gleichung ist Hooke's Gesetz für normale Dehnung in x-Richtung eines Feststoffteilchens, welches einer Spannung, gemäß nachfolgender Abbildung unterliegt

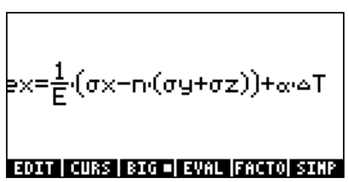
$$\begin{bmatrix} \sigma_{xx} & \sigma_{xy} & \sigma_{xz} \\ \sigma_{yx} & \sigma_{yy} & \sigma_{yz} \\ \sigma_{zx} & \sigma_{zy} & \sigma_{zz} \end{bmatrix}$$

Die Gleichung ist $e_{xx} = \frac{1}{E}[\sigma_{xx} - n \cdot (\sigma_{yy} + \sigma_{zz})] + \alpha \cdot \Delta T,$ wobei e_{xx} die Einheit der Dehnung in x-Richtung darstellt, σ_{xx}, σ_{yy} und σ_{zz}, die normale Spannung auf die Teilchen in Richtung der Achsen x, y und z, *E* ist Young's Elastizitätsmodul des Materials, *n* ist das Poisson-Verhältnis des Materials, α der thermische Dehnungskoeffizient des Materials und ΔT der Temperaturanstieg ist.

Angenommen Sie haben folgende Daten: σ_{xx}= 2500 psi, σ_{yy}=1200 psi, und σ_{zz} = 500 psi, E = 1200000 psi, *n* = 0.15, α = 0,00001/°F, ΔT = 60 °F. Um die Dehnung e_{xx} zu berechnen, gehen Sie wie folgt vor:

[→] NUM.SLV [OK]	Starten Sie den numerischen Löser, um die Gleichung zu lösen
[→] EQW	Starten Sie den EquationWriter, um die Gleichung einzugeben

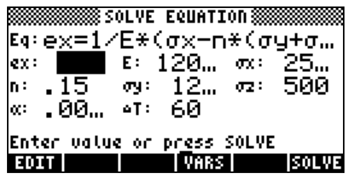
An dieser Stelle befolgen Sie die Anweisungen aus Kapitel 2, Verwendung des EquationWriters zur Erstellung einer Gleichung. Die Gleichung, die Sie ins Feld *Eq* eingeben, sollte so aussehen (beachten Sie, dass wir nur einen Unterindex benutzen, um auf die Variablen hinzuweisen, d.h. e_{xx} wird als *ex* übertragen, usw. -- dies wird gemacht, um Zeit beim eintippen zu sparen):



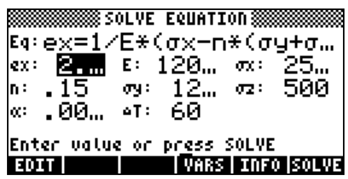
Benutzen Sie folgende Abkürzungen für Sonderzeichen:

σ: [ALPHA] [→] [S]   α: [ALPHA] [→] [A]   Δ: [ALPHA] [→] [C] beachten Sie dabei, dass die Kleinbuchstaben mit ALPHA ← vor dem Buchstaben eingegeben werden, somit wird x als ALPHA ← X eingegeben.

Drücken Sie ENTER, um zum Löser zurückzukehren. Geben Sie die oben vorgeschlagenen Werte in die entsprechenden Felder ein, so dass die Anzeige im Löser wie folgt aussieht:



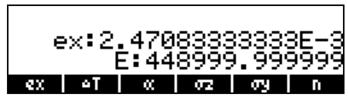
Mit hervorgehobenem *ex:* Feld, drücken Sie SOLVE, um die Lösung für *ex:* zu ermitteln.



Drücken Sie EDIT, während das Feld *ex*: markiert ist, ist die Lösung in der SOLVE EQUATION Eingabemaske zu sehen. Das Ergebnis ist 2,470833333333E-3. Drücken Sie OK, um EDIT (Bearbeitungsmodus) zu verlassen.

Angenommen Sie möchten nun das Young Modul, welches die Dehnung von e_{xx} = 0,005, unter der gleichen Spannung erzeugt, wobei die thermische Ausdehnung unbeachtet bleibt. In diesem Fall, sollten Sie den Wert 0,005 in das Feld *ex*: eingeben und eine Null in das Feld *ΔT:* (ist ΔT = 0, werden keine thermischen Effekte berücksichtigt). Um E zu berechnen, markieren Sie das Feld *E*: und drücken Sie SOLVE. Das Ergebnis, mit der EDIT Funktion angesehen, ist E ≈449000 psi. Drücken Sie SOLVE ENTER um zur Normalanzeige zurückzukehren.

Beachten Sie, dass die Ergebnisse der Berechnungen, die innerhalb des numerischen Lösers stattfinden, in den Stack kopiert wurden:



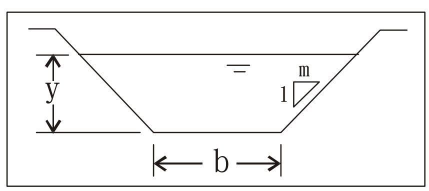
Auch werden Sie in Ihren Funktionstasten die Variablen, die den, die Sie in der Gleichung, EQ gespeichert haben, entsprechen, finden (drücken Sie NXT, um alle Variablen in Ihrem Verzeichnis anzuzeigen), d.h. die Variablen *ex*, *ΔT*, *α*, *σz*, *σy*, *n*, *σx* und *E*.

Beispiel 2 – Spezifische Leistung im offen Kanalsdurchfluss

Spezifische Leistung in einem offenen Kanaldurchfluss wird, als Leistung pro gemessene Gewichtseinheit zum Kanalboden, bezeichnet. Nehmen wir an E = spezifische Leistung, y = Kanaltiefe, V = Durchflussgeschwindigkeit, g = Gravitätsbeschleunigung, dann schreiben wir

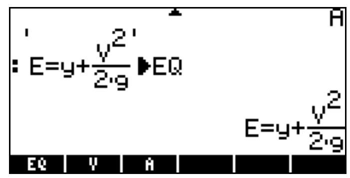
$$E = y + \frac{V^2}{2g}.$$

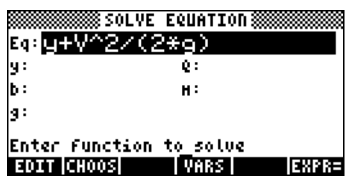
Die Durchflussgeschwindigkeit ist durch V = Q/A gegeben, wobei Q = Wasserabfluss, A = die Fläche des Querschnitts darstellt. Die Fläche ist abhängig vom verwendeten Querschnitt, z.B. für einen trapezförmigen Querschnitt, wie in Abbildung unten gezeigt, A = (b+m·y) ·y, wobei b = die Breite des Kanalbodens und m = Seitenwandneigung des Querschnitts ist.



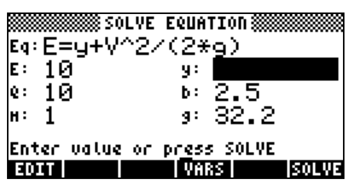
Wir können E in die Gleichung, wie oben gezeigt und zusätzlich die Variablen A und V eingeben, so dass die Eingabemaske für die

Grundvariablen y, Q, g, m und b, wie nachfolgend gezeigt, Eingabefelder zur Verfügung stellt:

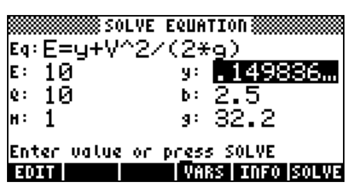
- Erstellen wir zuerst ein Unterverzeichnis SPEN (SPecific ENergy - spezifisiche Leistung) und arbeiten in diesem Unterverzeichnis.
- Als nächstes definieren Sie folgende Variablen:
-  

- Starten Sie den numerischen Löser, um die Gleichungen zu lösen: NUM.SLV OK. Beachten Sie, dass die Eingabemaske bereits Einträge für die Variablen y, Q, b, m und g enthält:
- 

- Versuchen Sie folgende Eingabedaten: E = 10 ft, Q = 10 cfs (Kubikfuß pro Sekunde), b = 2,5 ft, m = 1,0, g = 32,2 ft/s^2:

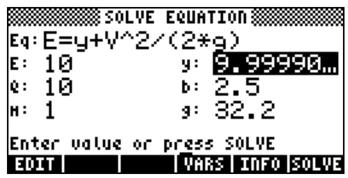


- Lösen Sie die Gleichung für y.



Das Ergebnis ist 0,149836.., d.h., y = 0,149836.

- Es ist aber bekannt, dass es für y in dieser Gleichung für spezifische Leistung eigentlich zwei Lösungen gibt. Die Lösung, die wir gerade ermittelt haben entspricht einer numerischen Lösung mit einem Ausgangswert von 0 (der Standardwert für y, d.h. wann immer das Lösungsfeld leer ist, ist der Ausgangswert 0). Um die andere Lösung zu finden, müssen wir einen größeren Wert für y eingeben, sagen wir 15, markieren anschließend das Eingabefeld y und lösen Sie die Gleichung für y ein weiteres Mal:
-



Das Ergebnis ist 9,99990, d.h., y = 9,99990 ft.

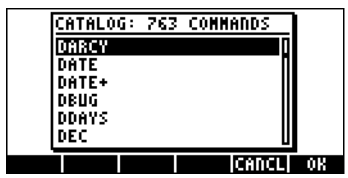
Dieses Beispiel veranschaulicht die Anwendung zu zusätzlichen Variablen zur Erstellung komplizierter Gleichungen. Sobald NUM.SLV aktiviert ist, werden die von den zusätzlichen Variablen implizierten Ersetzungen eingefügt und die Eingabemaske für die Gleichung stellt ein Eingabefeld für primitive oder fundamentale Variablen, die aus der Ersetzung resultieren, zur Verfügung. Das Beispiel veranschaulicht auch, eine Gleichung die mehr als eine Lösung hat und wie man den ersten Schätzwert für die Lösung wählt, welcher verschiedene Lösungen ermitteln kann.

Im nächsten Beispiel werden wir die Funktion DARCY zur Berechnung des Reibungsfaktors in Rohrleitungen verwenden. Dafür definieren wir die Funktion im folgenden Rahmen.

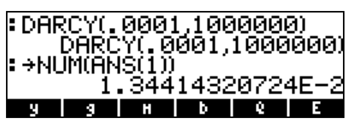
Sonderfunktion für Rohrleitungsdurchfluss: DARCY (ε/D,Re)

Die Darcy-Weisbach Gleichung wird zur Berechnung des Leistungsverlustes (pro Gewichtseinheit), h_f, bei einem Rohrleitungsdurchmesser D, absolute Rauheit ε und Länge L, wenn die Durchflussgeschwindigkeit in der Leitung V ist, verwendet. Die Gleichung lautet $h_f = f \cdot \frac{L}{D} \cdot \frac{V^2}{2g}$. Die Menge f wird als Reibungsfaktor des Durchflusses bezeichnet und wurde aus der Funktion der relativen Rauheit der Rohrleitung, ε/D und einer (dimensionslosen) Rynoldschen Zahl Re, berechnet. Die Reynoldsche Zahl wird als Re = ρVD/μ = VD/ν definiert, wobei ρ und μ die Dichte und dynamische Viskosität der Flüssigkeit entsprechend darstellen, während ν = μ/ρ die kinematische Viskosität der Flüssigkeit.

Der Rechner stellt eine Funktion mit dem Namen DARCY, welche als Eingabe die relative Rauheit ε/D und die Reynoldsche Zahl, in dieser Reihenfolge, zur Berechnung des Reibungsfaktors f hat, zur Verfügung. Die Funktion DARCY kann über den Befehle Katalog aufgerufen werden:



So z.B, können Sie für ε/D = 0,0001, Re = 1000000 den Reibungsfaktor berechnen, indem Sie wie folgt eingeben: DARCY(0,0001,1000000). In der nachfolgenden Abbildung wurde die Funktion →NUM () zur Berechnung des numerischen Wertes der Funktion verwendet:



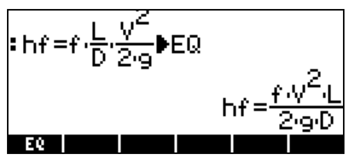
Das Ergebnis ist f = DARCY(0,0001,1000000) = 0,01341…

Die Funktion FANNING(ε/D,Re)

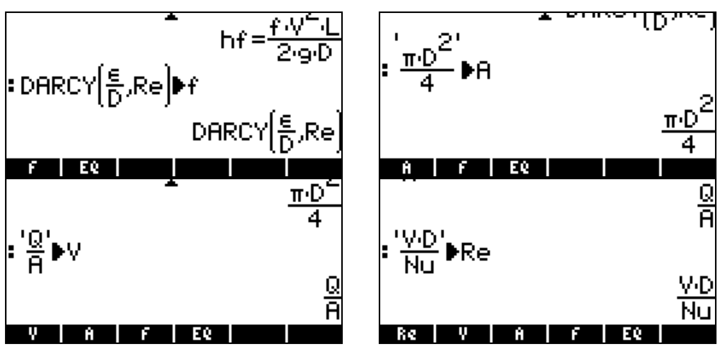
In aerodynamischen Anwendungen wird ein anderer Reibungsfaktor verwendet, der sogenannte Fanning Reibungsfaktor. Der Fanning Reibungsfaktor f_F, wird als der 4-fache Darcy-Weisbach Reibungsfaktor *f,* bezeichnet. Der Rechner stellt eine Funktion FANNING zur Verfügung, welche die gleichen Eingaben wie die Funktion DARCY benutzt, d.h. ε/D und Re, und stellt den FANNING Reibungsfaktor als Ergebnis zur Verfügung. Überprüfen Sie, ob FANNING(0,0001,1000000) = 0,0033603589181s, ergibt.

Beispiel 3 – Strömung in einem Rohr

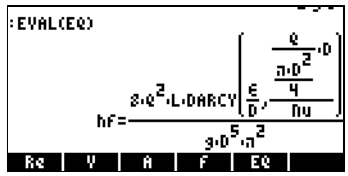
Für die nachfolgenden Beispiele sollten Sie eine separates Unterverzeichnis (PIPES) erstellen. Die Hauptgleichung für die Strömung in einem Rohr, ist, wie kann es auch anders sein, die Darcy-Weisbach Gleichung. Geben Sie also nachfolgende Gleichung in EQ ein:



Genauso die folgenden Variablen (f, A, V, Re):



In diesem Fall haben wir die Hauptgleichung (Darcy-Weisbach Gleichung) in EQ gespeichert und anschließend alle ihre Variablen durch andere Ausdrücke, über die Definition der Variablen f, A, V und Re, ausgetauscht. Um die kombinierte Gleichung zu sehen, benutzen Sie EVAL(EQ). In diesem Beispiel haben wir die Anzeige so geändert, dass die gesamte Gleichung im Display angezeigt wird:

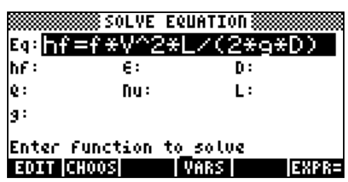


Somit lautet die zu lösende Gleichung, nachdem diese mit den verschiedenen Variablen aus dem Verzeichnis kombiniert wurde, wie folgt:

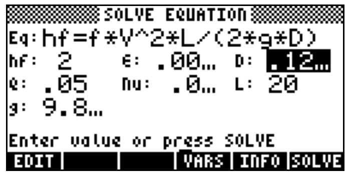
$$h_f = \frac{8Q^2 L}{\pi^2 g D^5} \cdot DARCY\left(\frac{\varepsilon}{D}, \frac{\frac{QD}{\pi D^2/4}}{Nu} \right)$$

Die kombinierte Gleichung enthält die primitiven Variablen *h_f, Q, L, g, D, ε* und *Nu*.

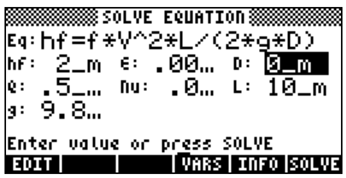
Starten Sie den numerischen Löser (⇨ NUM.SLV OK), um die in der SOLVE EQUATION Eingabemaske vorhandenen primitiven Variablen anzuzeigen.

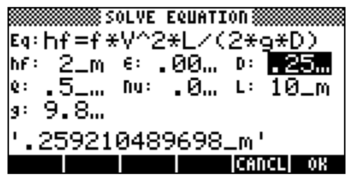


Angenommen wir verwenden die Werte hf = 2 m, ε = 0,00001 m, Q = 0,05 m^3/s, Nu = 0,000001 m^2/s, L = 20 m und g = 9,806 m/s^2 und wollen den Durchmesser D berechnen. Tragen Sie die Eingabewerte ein und lösen Sie die Gleichung für D, das Ergebnis lautet: 0,12, d.h., D = 0,12 m.



Ist die Gleichung dimensional gesehen konsistent, können Sie Einheiten zu den Eingabewerten, wie in der unteren Abbildung gezeigt, hinzufügen. Sie müssen diese Einheiten aber zu den ursprünglichen Schätzwerten in der Lösung hinzufügen. Im nachstehenden Beispiel fügen wir vor Lösung des Problems 0_m ins Feld D: ein. Die Lösung ist auf der rechten Abbildung zu sehen:



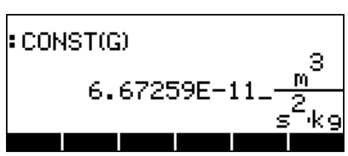


Drücken Sie die Taste ENTER, um zur Normalanzeige des Rechners zurückzukehren. Die Lösung für D wird im Stack angezeigt.

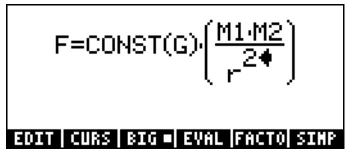
Beispiel 4 – Universelle Gravitation

Newtons Gesetz der universellen Gravitation sagt aus, dass die Magnitude der Anziehungskraft zweier Körper der Masse m_1 und m_2, die in einem Abstand *r* voneinander entfernt sind, über die Gleichung $F = G \cdot \frac{M_1 \cdot M_2}{r^2}$. dargestellt wird.

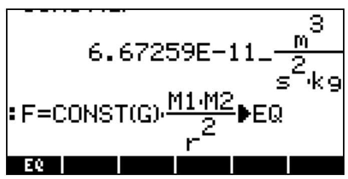
In diesem Fall ist G die universelle Gravitationskonstante, deren Wert man über die Funktion CONST im Rechner erhalten kann.



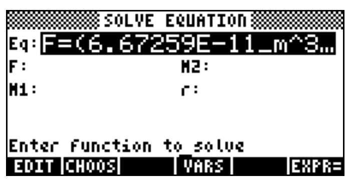
Wir können die Gleichung für jedes Glied, ausgenommen G, lösen, indem wir die Gleichung wie folgt eingeben:



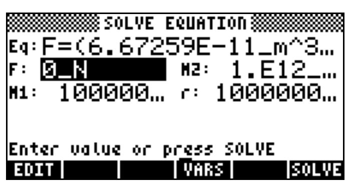
Diese Gleichung speichern wir dann in EQ:



Wenn Sie nun den numerischen Löser für diese Gleichung starten, erhalten Sie eine Eingabemaske mit den folgenden Eingabefeldern F, G, m1, m2 und r.



Lösen wir dieses Problem nun, indem wir verschiedene Einheiten für die bekannten Variablen einsetzen m1 = 1.0×10^{6} kg, m2 = 1.0×10^{12} kg, r = 1.0×10^{11} m. Geben Sie einen Wert von 0_N in Feld F ein, um sicher zu gehen, dass die Lösung richtig, mit den Einheiten des Rechners, ausgewertet wird:

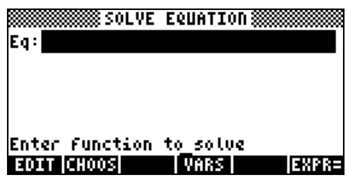


Lösen Sie die Gleichung für F, drücken Sie dann, um zur Normalanzeige zurückzukehren. Die Lösung ist F: 6,67259E-15_N, oder F = $6{,}67259\times10^{-15}$ N.

Anmerkung: Wenn Sie Einheiten im numerischen Löser benutzen, stellen Sie sicher, dass alle Variablen die richtigen Einheiten haben, dass diese kompatibel sind und die Gleichung dimensional gesehen homogen ist.

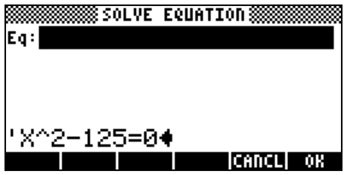
Unterschiedliche Wege Gleichungen in EQ einzugeben

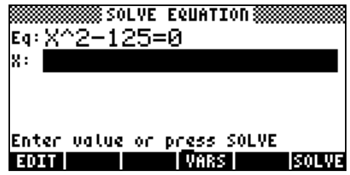
In all den gezeigten Beispielen, haben wir die zu lösenden Gleichung, direkt in die Variable EQ, vor Aktivierung des numerischen Lösers, eingegeben. Eigentlich kann die zu lösende Gleichung direkt in den Löser, nachdem Sie diesen gestartet haben, eingegeben werden, durch bearbeiten der Inhalte des Feldes EQ in der Eingabemaske des numerischen Lösers. Sollte die Variable EQ beim Start des numerischen Lösers (↱ NUM.SLV OK) noch nicht definiert sein, wird das Feld EQ hervorgehoben:



Sie können an dieser Stelle entweder eine neue Gleichung, über EDIT eingeben. Sie bekommen ein Apostrophenpaar, so dass Sie den Ausdruck dazwischen eingeben können:

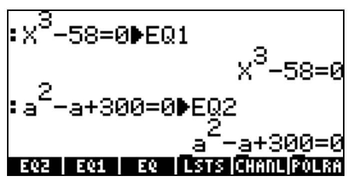
Geben Sie die Gleichung, sagen wir X^2 - 125 = 0 direkt in den Stack ein, müssen rücken Sie dann OK drücken.





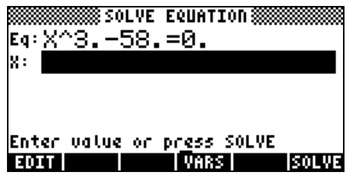
An dieser Stelle ist die Gleichung zur Lösung bereit.
Alternativ dazu, können Sie den EquationWriter starten, nachdem Sie EDIT gedrückt haben, um ihre Gleichung zu bearbeiten. Drücken Sie ENTER, um zum EquationWriter zurückzukehren.

Eine weitere Möglichkeit eine Gleichung in die Variable EQ einzugeben ist eine bereits bestehende Variable, die in EQ eingegeben werden soll, aus dem Verzeichnis auszuwählen. Das bedeutet, dass Ihre Gleichung bereits in einer Variablen gespeichert sein muss. So z.B. nehmen wir an, dass wir die folgenden Gleichungen in die Variablen EQ1 und EQ2 bereits eingegeben haben:



Starten Sie nun den numerischen Löser (↱ NUM.SLV OK) und heben Sie das Feld EQ vor. Drücken Sie an dieser Stelle die Funktionstaste CHOOS. benutzen Sie die Pfeiltasten (▲▼), um, sagen wir die Variable EQ1, auszuwählen.

Nachdem Sie die Variable EQ1 ausgewählt haben, drücken Sie OK, um die Variable EQ in den Löser zu laden. Die neue Gleichung steht nun zur Lösung bereit.

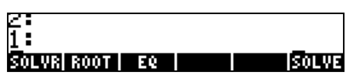


Das Funktionsmenü SOLVE

Über das Menü SOLVE kann über Funktionstasten auf einige der Funktionen des numerischen Lösers zugegriffen werden. Um in dieses Menü im RPN-Modus zu gelangen, verwenden Sie 74 MENU, oder im ALG-Modus: MENU(74) Alternativ dazu, können Sie aber auch die Tastenkombination (halten) 7 zum starten des Menüs SOLVE benutzen. Die von SOLVE zur Verfügung gestellten Untermenüs lauten wie folgt:

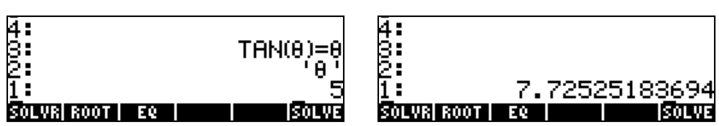
Das Untermenü ROOT

Im Untermenü ROOT sind folgende Funktionen und Untermenüs enthalten:

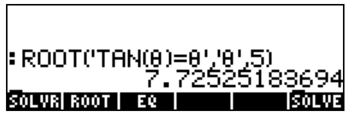


Die Funktion ROOT

Die Funktion ROOT wird zur Lösung einer Gleichung für eine gegebene Variable mit einem geschätzten Anfangswert verwendet. Im RPN-Modus befindet sich die Gleichung in Stack Ebene 3, während der Variablen Name in Ebene 2 zu finden ist und der geschätzte Anfangswert in Ebene 1. Die nachfolgende Abbildung zeigt den RPN-Stack vor und nach Aktivierung der Funktion ROOT:



Im ALG-Modus würden Sie zum Starten der Funktion ROOT wie folgt vorgehen: ROOT(‘TAN(θ)=θ’,’θ’,5)



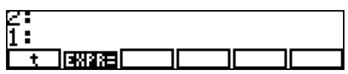
Variable EQ

Die Funktionstaste EQ in diesem Untermenü wird als Referenz zur Variablen EQ verwendet. Das Drücken der Funktionstaste ist gleichwertig mit dem Verwenden der Funktion RCEQ (ReCall EQ).

Das Untermenü SOLVER

Das Untermenü SOLVR startet das Funktionsmenü Löser für die momentan in EQ gespeicherte Gleichung. Nachfolgend einige Beispiele:

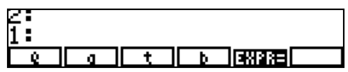
<u>Beispiel 1</u> – Lösen der Gleichung t^2-5t = -4
So z.B., wenn Sie die Gleichung ‘t^2-5*t=-4’ in EQ speichern, und anschließend SOLVR drücken, wird nachfolgendes Menü gestartet:



Dieses Ergebnis zeigt an, dass Sie die Gleichung, am oberen Rand des Displays, für einen Wert t, lösen können. Wenn Sie z.B. versuchen ← [t] zu lösen, erhalten Sie den Wert t: 1, nachdem kurz die Meldung "Solving for t" (t lösen) aufblinkt. Es gibt aber eine weitere Wurzel für diese Gleichung, welche man durch Änderung des Wertes von t, bevor man dieses erneut berechnet, erhält. Machen Sie folgendes: 10 [t], drücken Sie dann ← [t]. Die Lösung lautet nun t: 4,0000000003. Um dieses Ergebnis zu überprüfen, drücken Sie die Funktionstaste EXPR=, welche den Ausdruck in EQ mit dem aktuellen Wert von t, berechnet. In diesem Fall ist das Ergebnis:

Um die SOLVR Umgebung zu verlassen, drücken Sie VAR. An dieser Stelle haben Sie keinen Zugang zum Menü SOLVE, somit müssen Sie dieses erneut, wie oben angezeigt, starten, um mit den nachfolgenden Beispielen fortzufahren.

Beispiel 2 – Lösen der Gleichung Q = at^2+bt
In EQ kann auch Gleichung mit mehr als einer Variablen gespeichert werden, nehmen wir an es ist die Gleichung 'Q = at^2 + bt'. Wenn Sie das Funktionsmenü SOLVE in diesem Fall gestartet haben und ROOT SOLVR drücken, erhalten Sie nachfolgende Anzeige:

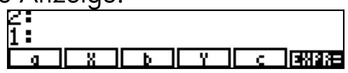


In dieser SOLVR Umgebung können Sie Werte für jede einzelne aufgelistete Variable festlegen, indem Sie diesen Wert einfach in den Stack eingeben und die entsprechende Funktionstaste drücken. Angenommen wir wollen die Werte Q = 14, a = 2 und b = 3 eingeben. Sie würden es folgendermaßen tun: 14 [Q], 2 [a], 3 [b].

Da den Variablen Q, a und b numerische Werte zugewiesen werden, erscheinen die zugewiesenen Werte in der linken oberen Ecke des Displays. An dieser Stelle können wir t mit Hilfe von ⬅ [t] lösen. Das Ergebnis ist t: 2. Drücken Sie nun EXPR= erhalten Sie die Ergebenisse:

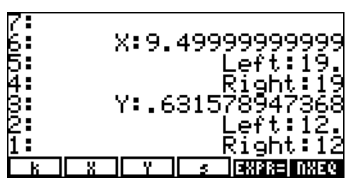
Beispiel 3 – Zwei simultane Gleichungen lösen, jeweils eine auf einmal
Sie können mehr als eine Gleichung lösen, indem Sie erst eine Gleichung lösen und den gleichen Vorgang solange wiederholen, bis eine Lösung gefunden wurde. So z.B., wenn Sie nachfolgende Liste von Gleichungen in die Variable EQ eingeben - EQ: { 'a*X+b*Y = c', 'k*X*Y=s'},

erhalten Sie mit der Tastenfolge ROOT SOLVE, innerhalb des Funktionsmenüs SOLVE die nachfolgende Anzeige:

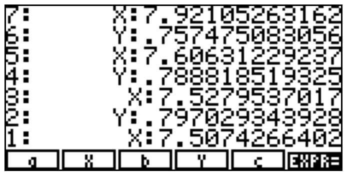


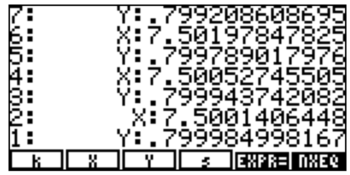
Die erste Gleichung, d.h. a*X + b*Y = c wird im oberen Teil des Displays angezeigt. Sie können Werte für die Variablen a, b und c eingeben, sagen wir:2 [a] 5 [b] 19 [c]. Da wir nur eine Gleichung auf einmal lösen können, geben wir einen geschätzten Anfangswert für Y, sagen wir 0 [Y] ein und lösen die Gleichung für X, mit ← [X]. Dies ergibt den Wert X: 9,4999…. Um den Wert der Gleichung an dieser Stelle zu überprüfen, drücken Sie EXPR=. Die Ergebnisse sind: Left (links): 19, Right (rechts): 19. Um die nächste Gleichung zu lösen, drücken Sie NXT NXEQ. Im Display werden die Funktionstasten wie folgt angezeigt:

Geben wir z.B. die Werte k = 2, s = 12 ein. Lösen Sie die Gleichung dann für Y und drücken Sie EXPR=. Die Ergebnisse sind nun, Y:



Dann fahren wir fort und navigieren wir zwischen den beiden Gleichungen, vor und zurück, lösen die erste für X und die zweite für Y, solange bis X und Y zu einer Lösung führen. Um sich zwischen den Gleichungen hin und her zu bewegen verwenden wir die Taste NXEQ. Um X und Y zu lösen, benutzen Sie entsprechend ← [X] und ← [Y]. Die nachfolgende Sequenz von Lösungen wird erstellt:





Nachdem Sie nun die beiden Gleichungen gelöst haben, jeweils eine auf einmal, bemerken wir, dass X bis zur dritten Nachkommastelle, sich dem Wert 7,500, während Y sich dem Wert 0,799 nähert.

Verwenden der Maßeinheiten mit dem SOLVR Unterprogramm

Diese sind einige Richtlinien auf dem Gebrauch von Maßeinheiten mit dem SOLVR Unterprogramm:

- Das Eintragen einer Vermutung mit Maßeinheiten für eine gegebene Variable, stellt den Gebrauch von jenen Maßeinheiten in der Lösung vor.
- Wenn eine neue Vermutung ohne Maßeinheiten gegeben wird, werden die Maßeinheiten, die vorher für die bestimmte Variable gespeichert werden, benutzt.
- Um Maßeinheiten zu entfernen tragen Sie eine Zahl ohne Maßeinheiten in eine Liste als die neue Vermutung ein, d.h. verwenden Sie das Format {Zahl}.
- Eine Liste von Zahlen kann als Vermutung für eine Variable gegeben werden. In diesem Fall nimmt die Maßeinheiten die benutzten Maßeinheiten gehören der letzten Zahl in der Liste. Z.B. zeigt das Hereinkommen {1.41_ft 1_cm 1_m} an, daß Meßinstrumente (M) für diese Variable benutzt werden.
- Der Ausdruck, der in der Lösung verwendet wird, muß konsistente Glieder haben, oder eine Störung resultiert beim Versuchen, für einen Wert zu lösen.

Das Untermenü DIFFE

Das Untermenü DIFFE enthält eine Reihe von Funktionen für die numerische Lösung von Differenzialgleichungen. Die zur Verfügung stehenden Funktionen sind:

Diese Funktionen werden im Detail in Kapitel 16 erläutert.

Das Untermenü POLY

Das Untermenü POLY führt Operationen mit Polynomen durch. Die enthaltenen Funktionen sind:

Funktion PROOT

Diese Funktion wird dazu verwendet die Wurzeln eines Polynoms mit bekanntem Vektor, mit den Koeffizienten des Polynoms in absteigender Reihenfolge der Potenz der unabhängigen Variable, zu ermitteln. Mit anderen Worten, wenn das Polynom $a_nx^n + a_{n-1}x^{n-1} + \ldots + a_2x^2 + a_1x + a_0$, ist, sollte der Vektor von Koeffizienten als $[a_n, a_{n-1}, \ldots , a_2, a_1 , a_0]$ eingegeben werden. So sind z.B. die Wurzeln des Polynoms mit den Koeffizienten [1, -5, 6] die Werte [2, 3].

Funktion PCOEF

Diese Funktion erzeugt die Koeffizienten $[a_n, a_{n-1}, \ldots , a_2, a_1 , a_0]$ eines Polynoms $a_nx^n + a_{n-1}x^{n-1} + \ldots + a_2x^2 + a_1x + a_0$, mit bekannten Wurzeln eines Vektors $[r_1, r_2, \ldots, r_n]$. So z.B. ein Vektor dessen Wurzeln [-1, 2, 2, 1, 0] sind, ergbit folgende Koeffizienten: [1, -4, 3, 4, -4, 0] Das Polynom lautet $x^5 - 4x^4 + 3x^3 + 4x^2 - 4x$.

Funktion PEVAL

Diese Funktion wertet ein Polynom, mit bekanntem Vektor seiner Koeffizienten $[a_n, a_{n-1}, \ldots , a_2, a_1 , a_0]$ und einem Wert von x_0 aus, d.h., PEVAL berechnet $a_nx_0^n + a_{n-1}x_0^{n-1} + \ldots + a_2x_0^2 + a_1x_0 + a_0$. So z.B. ergibt die Funktion PEVAL den Wert 28, für die Koeffizienten [2, 3, -1, 2] und den Wert 2.

Das Untermenü SYS

Das Untermenü SYS enthält eine Auflistung von Funktionen zur Lösung linearer Systeme. Die in diesem Untermenü aufgelisteten Funktionen sind:

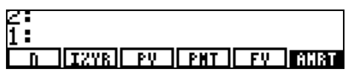
Diese Funktionen werden im Detail in Kapitel 11 erläutert.

Das Untermenü TVM

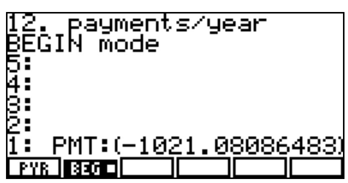
Das Untermenü TVM enthält Funktionen zur Berechnung des Geldzeitwertes (Time Value of Money). Dies ist eine alternative Möglichkeit FINANCE Probleme zu lösen (siehe Kapitel 6). Die Funktionen werden nachfolgend angezeigt:

Das Untermenü SOLVR

Das Untermenü SOLVR aus dem Untermenü TVM startet den Löser zur Lösung von TVM Problemen. Wenn Sie an dieser Stelle SOLVR drücken, erhalten Sie die nachfolgende Anzeige:



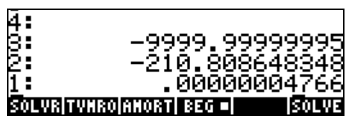
Als Beispiel versuchen wir die Werte n = 10, I%YR = 5,6, PV = 10000 und FV = 0 und geben dann ← [PMT] um PMT = -1021,08…. zu berechnen, ein. Wenn Sie nun NXT drücken, erscheint folgende Anzeige:



Drücken Sie VAR, um die SOLVR Umgebung zu verlassen. Suchen Sie den Weg zurück zum Untermenü TVM innerhalb des Untermenüs SOLVE, um auch die anderen vorhandenen Funktionen zu testen.

Funktion TVMROOT
Diese Funktion benötigt als Argument den Namen einer der Variablen im TVM Problem. Die Funktion gibt die Lösung für diese Variable zurück, vorausgesetzt die anderen Variablen existieren und die Werte dieser wurden vorher gespeichert. So z.B., nachdem wir eines der obigen TVM Probleme gelöst haben, können wir, sagen wir für 'N' wie folgt lösen: ['] ALPHA N ENTER TVMRO. Das Ergebnis ist 10.

Funktion AMORT
Diese Funktion nimmt einen Wert, die einen Zahlungszeitraum darstellt (zwischen 0 und n) und gibt den Hauptwert, die Zinsrate und den Saldo für die, zu dem Zeitpunkt gespeicherten, TVM Variablen zurück. Wenn wir z.B. mit den vorhin benutzten Daten, die Funktion AMORT, für einen Wert 10 starten, bekommen wir:



Funktion BEG
Wenn ausgewählt, werden die Berechnungen innerhalb der Funktion TVM an den Anfang jeder Zahlungsperiode gelegt. Wenn nicht ausgewählt, werden die Berechnungen innerhalb der Funktion TVM ans Ende jeder Zahlungsperiode gelegt.

Kapitel 7
Lösen von Mehrfachgleichungen

Viele wissenschaftliche und technische Probleme benötigen die gleichzeitige Lösung mehrerer Gleichungen. Der Rechner stellt mehrere Verfahrensweisen zur Lösung von Mehrfachgleichungen , wie unten gezeigt, zur Verfügung. Beachten Sie, dass keine Lösungen für System mit linearen Gleichungen in diesem Kapitel vorgestellt werden. Lösungen für lineare System werden in einem späteren Kapitel über Matrizen und lineare Algebra ausführlich erklärt.

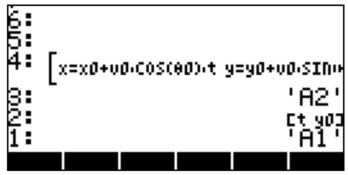
Rationelle Gleichungssysteme

Gleichungen die sich in Polynome oder rationelle algebraische Ausdrücke umwandeln lassen, können mit der Funktion SOLVE direkt mit dem Rechner gelöst werden. Sie müssen die Liste der Gleichungen als Elemente eines Vektors zur Verfügung stellen. Die Liste der zu lösenden Variablen, muss ebenfalls als Vektor zur Verfügung stehen. Bevor Sie versuchen mit dieser Prozedur eine Lösung zu ermitteln, stellen Sie sicher, dass das CAS im Exakt-Modus ist. Auch sollten Sie beachten, dass, je komplizierter ein Ausdruck ist, desto länger benötigt das CAS zur Lösung eines bestimmten Systems von Gleichungen. Beispiele für diese Anwendung finden Sie im Anschluss:

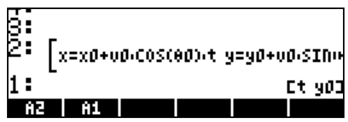
Beispiel 1 – Projektilbewegung

Verwenden Sie die Funktion SOLVE mit nachfolgenden Vektorargumenten, wobei das erste Argument die Liste der Gleichungen ['x = x0 + v0*COS(θ0)*t' 'y =y0+v0*SIN(θ0)*t – g*t^2/2']`ENTER`, darstellt, das zweite hingegeben die zu lösenden Variablen, sagen wir t und y0, d.h., ['t' 'y0'].

In diesem Fall verwenden wir den RPN-Modus zur Lösungsfindung. Der einzige Grund dafür ist, dass wir die Lösung Schritt für Schritt berechnen können. Im ALG-Modus ist die Lösung ziemlich ähnlich. Zuerst speichern wir den ersten Vektor (Gleichungen) in eine Variable A2 und den Vektor mit den Variablen in die Variable A1. Im RPN Stack sieht die Anzeige, bevor Sie die Variablen gespeichert haben, wie folgt aus:



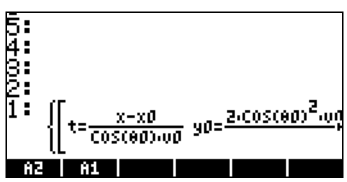
An dieser Stelle, müssen wir nur noch STO▶ zweimal drücken, um diese Variablen zu speichern.
Für die Lösungsfindung, wechseln Sie das CAS in den Exakt Modus, listen Sie dann die Inhalte der Variablen A2 und A1 – in dieser Reihenfolge - A2 A1 .

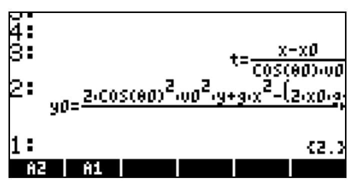


Verwenden Sie nun den Befehl SOLVE (aus dem Menü S.SLV: ↰ S.SLV) .
Nach etwa 40 Sekunden, vielleicht auch länger, erhalten Sie als Ergebnis eine Liste:

{ ‘t = (x-x0)/(COS(θ0)*v0)’
‘y0 = (2*COS(θ0)^2*v0^2*y+(g*x^2(2*x0*g+2*SIN(• 0))*COS(θ0)*v0^2)*x+
(x0^2*g+2*SIN(θ0)*COS(θ0)*v0^2*x0)))/(2*COS(θ0)^2*v0^2)’]}

Drücken Sie EVAL um den Vektor aus der Liste zu entfernen, verwenden Sie anschließend de Befehl OBJ→, um die Gleichungen separat im Stack aufzulisten.





Anmerkung: Diese Methode hat in diesem Beispiel einwandfrei funktioniert, weil die Unbekannten t und y0 algebraische Ausdrücke der Gleichungen darstellten. Diese Methode funktioniert aber nicht, wenn wir versuchen θ0 zu lösen, weil θ0 ein transzendenter Terminus ist.

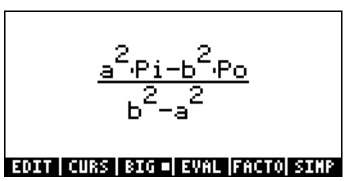
Beispiel 2 – Spannungen in einem dickwandigen Zylinder

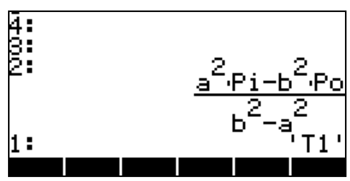
Nehmen wir an, wir haben einen dickwandigen Zylinder mit Innen- und Außendurchmesser *a* und *b*, entsprechend, welche einem inneren P_i und einem äußeren Druck P_o ausgesetzt sind. An jedem Radialabstand *r* von der Achse des Zylinders aus in Radial- und Querrichtung σ_{rr} and $\sigma_{\theta\theta}$ entsprechend, wird die Spannung durch folgende Formel berechnet:

$$\sigma_{\theta\theta} = \frac{a^2 \cdot P_i - b^2 \cdot P_o}{b^2 - a^2} + \frac{a^2 \cdot b^2 \cdot (P_i - P_o)}{r^2 \cdot (b^2 - a^2)},$$

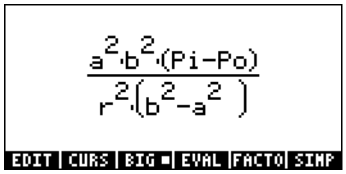
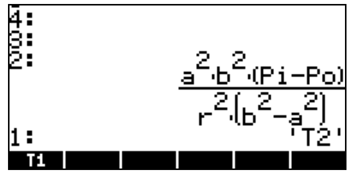
$$\sigma_{rr} = \frac{a^2 \cdot P_i - b^2 \cdot P_o}{b^2 - a^2} - \frac{a^2 \cdot b^2 \cdot (P_i - P_o)}{r^2 \cdot (b^2 - a^2)}.$$

Beachten Sie dabei, dass der einzige Unterschied zwischen den beiden Gleichungen in der rechten Seite der Ausdrücke zu finden ist und allein im Vorzeichen besteht. Um diese Gleichungen in den Rechner einzugeben, schlage ich vor, Sie geben den ersten Ausdruck ein und speichern diesen unter T1, dann den zweiten und speichern diesen unter T2. Eine spätere Eingabe dieser Gleichungen besteht dann nur im einladen von T1 und T2 in den Stack, um diese dann zu addieren und zu subtrahieren. Nachfolgend wird gezeigt, wie Sie dies im EquationWriter tun können.
Geben Sie T1 ein und speichern Sie den Ausdruck:

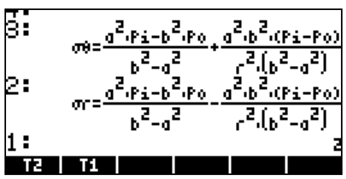
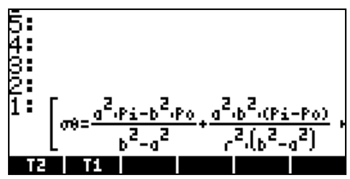




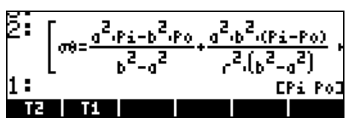
Geben Sie T2 ein und speichern Sie den Ausdruck:

Beachten Sie, dass wir in diesem Beispiel den RPN-Modus verwenden, die Vorgehensweise im ALG-Modus ist jedoch ziemlich ähnlich. Erstellen Sie die Gleichung für $\sigma_{\theta\theta}$: VAR T1 T2 + ALPHA ↱ S ALPHA ↱ T ENTER ▶ ↱ =
Erstellen Sie die Gleichung für σ_{rr}: VAR T1 T2 − ALPHA ↱ S ALPHA ↰ R ENTER ▶ ↱ =
Erstellen Sie aus diesen beiden Gleichungen einen Vektor mit der Funktion →ARRAY (starten Sie diese aus dem Befehle Katalog mit ↱ CAT) nachdem Sie eine 2 eingegeben haben.

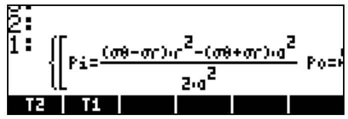



Nehmen wir nun an, dass wir P_i und P_o, mit vorgegebenen *a*, *b*, r, σ_{rr} und $\sigma_{\theta\theta}$ lösen wollen. Wir erstellen einen Vektor mit den Unbekannten:

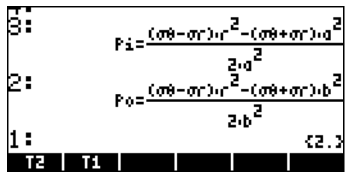


Zur Lösung für P_i und P_o verwenden wir den Befehl SOLVE aus dem Menü S.SLV (↰ S.SLV), es dauert etwa eine Minute bis das Ergebnis im Rechner vorliegt:

{[‘Pi=-(((σθ-σr)*r^2-(σθ+σr)*a^2)/(2*a^2))’
‘Po=-(((σθ-σr)*r^2-(σθ+σr)*b^2)/(2*b^2))’] }, d.h.,



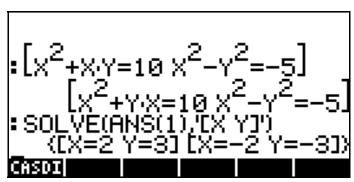
Beachten Sie, dass das Ergebnis einen Vektor [ ] innerhalb einer Liste { } darstellt. Benutzen Sie EVAL, um das Symbol für Liste zu entfernen.Verwenden Sie die Funktion OBJ →, um den Vektor zu zerlegen. Die Lösung lautet:



Diese beiden Beispiele stellen Systeme von linearen Gleichungen dar, welche genauso gut mit der Funktion LINSOLVE (siehe Kapitel 11) bearbeitet werden können. Das nachfolgende Beispiel zeigt die Funktion SOLVE angewendet auf ein System von Polynomgleichungen.

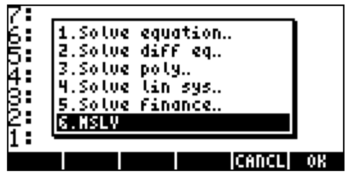
Beispiel 3 – System von Polynomgleichungen

Die nachfolgende Abbildung zeigt die Lösung des Systems $X^2+XY=10$, $X^2-Y^2=-5$ mit Hilfe der Funktion SOLVE:

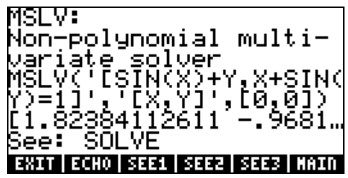


Lösungen zu simultanen Gleichungen mit MSLV

Die Funktion MSLV ist die letzte Option im Menü ↱ NUM.SLV :



Nachfolgend finden Sie den Hilfe-Eintrag für die Funktion MSLV:

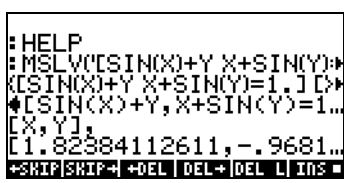


Beispiel 1 – Beispiel aus der Hilfefunktion

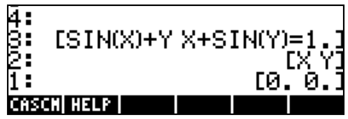
Wie mit allen anderen Einträgen für Funktionen, gibt es in der Hilfefunktion, auch ein Beispiel zum Eintrag MSLV, wie oben gezeigt. Beachten Sie, dass die Funktion MSLV drei Argumente benötigt:

1. Einen Vektor der die Gleichungen enthält, d.h. '[SIN(X)+Y,X+SIN(Y)=1]'
2. Einen Vektor der die zu lösenden Variablen enthält, d.h. '[X,Y]'
3. Einen Vektor der die ursprünglichen Werte für die Lösung beinhaltet, d.h. die ursprünglichen Werte beider Variablen X und Y sind in diesem Beispiel 0.

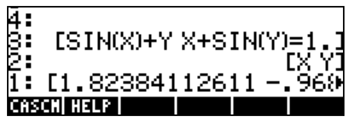
Im ALG-Modus drücken Sie ECHO , um das Beispiel in den Stack zu kopieren, drücken Sie dann ENTER, um das Beispiel auszuführen. Um alle Elemente der Lösung anzusehen, müssen Sie den Zeileneditor mit der Pfeiltaste (▼) aktivieren:



Im RPN-Modus wird die Lösung für dieses Beispiel wie folgt gefunden:



Durch Aktivierung der Funktion MSLV erscheint folgende Anzeige.



Sie haben wahrscheinlich festgestellt, dass in der linken oberen Ecke des Displays, während der Berechnung der Lösung, Zwischenergebnisse angezeigt werden. Da die von MSLV gelieferte Lösung numerisch ist, zeigen die Informationen in der linken oberen Ecke die Ergebnisse des wiederholenden Prozesses, auf dem Weg zur Lösung an. Die endgültige Lösung ist *X = 1,8238, Y = -0,9681*.

Beispiel 2 – Eingang aus einem See in einen offenen Kanal

Bei diesem speziellen Problem mit dem Durchfluss in einem offenen Kanal, muss die Lösung für zwei Gleichungen simultan erfolgen, die

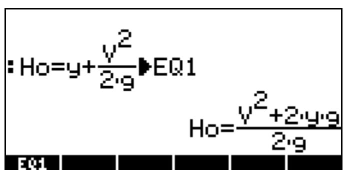
Energiegleichung: $H_o = y + \dfrac{V^2}{2g}$ und die Manning-Gleichung:

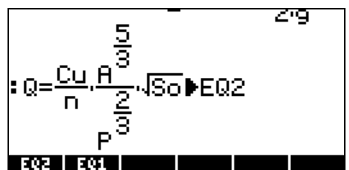
$Q = \dfrac{Cu}{n} \cdot \dfrac{A^{5/3}}{P^{2/3}} \cdot \sqrt{S_o}$. In diesen Gleichungen stellt H_o die Druckhöhe (m oder ft) dar, welche beim Durchfluss am Eingang des Kanals vorhanden ist, y die Durchflusstiefe (m oder ft), V = Q/A die Geschwindigkeit (m/s oder ft/s) dar, Q die Wassermenge (m^3/s or ft^3/s), A den Flächeninhalt des Querschnitts (m^2 oder ft^2), C_u den Koeffizienten, welcher von den Systemeinheiten (C_u = 1,0 für die SI, C_u = 1,486 für das englische Einheitensystem) abhängt, n der Manning-Koeffizient, eine Maßeinheit der Rauheit der Kanalwandoberfläche (z.B. für Beton, n = 0,012), P der benetzte Umfang des Querschnitts (m oder ft), S_o die Neigung des Kanalbettes als Bruch dargestellt. Für einen trapezförmigen Kanal, wie unten gezeigt, wird die Fläche über die Formel $A = (b + my)y$ dargestellt, während der benetzte

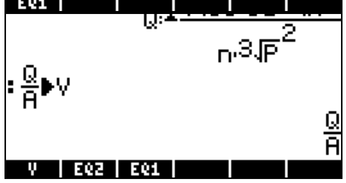
Umfang über $P = b + 2y\sqrt{1+m^2}$, wobei b die Breite des Bodens (m oder ft) und m die Seitenwandneigung (1V:mH) des Querschnittes darstellt.

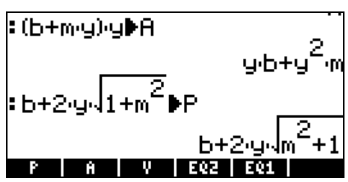
Normalerweise muss man die Energie- wie auch die Manning-Gleichung, simultan für y und Q lösen. Sobald diese Gleichungen als primitive Variablen b, m, y, g, S_o, n, Cu, Q und H_o dargestellt sind, bleibt uns das folgende Gleichungssystem $f_1(y,Q) = 0$, $f_2(y,Q) = 0$. Diese beiden Gleichungen können wir wie folgt erstellen.

Nehmen wir an, dass wir den ALG-Modus des Rechners benutzen, obwohl die Erstellung der Gleichungen und Lösen dieser mit MSLV im RPN-Modus ähnlich geht. Erstellen Sie ein Unterverzeichnis, sagen wir CHANL (für offener CHANneL – Kanal) und innerhalb dieses Unterverzeichnisses die nachfolgenden Variablen.



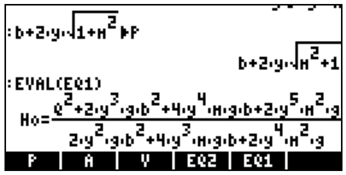


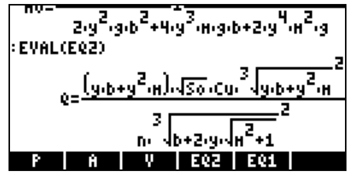




Um die ursprünglichen Gleichungen, EQ1 und EQ2, als einfache Variablen, wie oben angezeigt, zu sehen, können wir die Funktion EVAL nutzen, welche wir auf jede Gleichung anwenden, d.h., EVAL EQ1 EVAL EQ2. Die

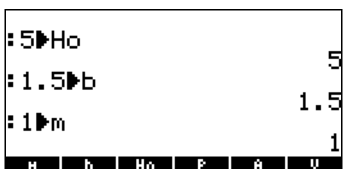
Gleichungen werden im Stack wie folgt angezeigt (kleine Schriftart ausgewählt):

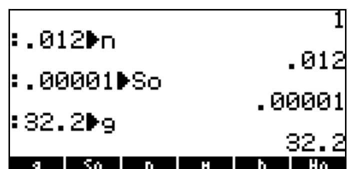


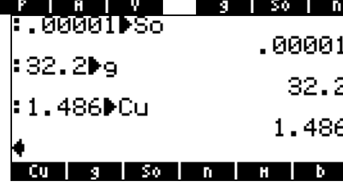


Wir stellen fest, dass diese Gleichungen tatsächlich als einfache Variable b, m, y, g, S_o, n, Cu, Q und H_o dargestellt werden können.

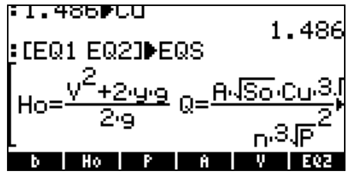
Um y und Q zu lösen, müssen wir den anderen Variablen Werte zuweisen. Angenommen wir verwenden folgende Werte: H_0 = 5 ft, b = 1,5 ft, m = 1, n = 0,012, S_0 = 0,00001, g = 32,2 und Cu = 1,486. Bevor wir diese Variablen zur Lösung mit MSLV verwenden können, müssen wir die Werte bei den entsprechenden Variablennamen eintragen. Dies kann wie folgt erreicht werden:



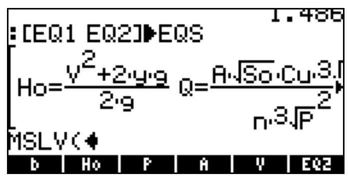




Nun sind wir bereit die Gleichung zu lösen. Vorher aber, müssen wir die beiden Gleichungen zusammen in einen Vektor geben. Dies erreichen wir, durch Speicherung des Vektors in einer Variablen mit dem Namen EQS (EQationS – Gleichungen):



Als Anfangswert für die Variablen y und Q verwenden wir y = 5 (entspricht dem Wert von H_o, welches der Maximalwert ist, den y annehmen kann) und Q = 10 (dies ist nur eine Schätzung). Um die Lösung zu erhalten, wählen wir die Funktion MSLV aus dem Menü NUM.SLV, z.B. [→] NUM.SLV [6] [OK], zur Eingabe des Befehls im Display:

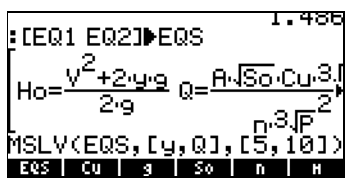


Als nächstes geben wir die Variable EQS [NXT] [NXT] [EQS], gefolgt vom Vektor [y,Q] ein:

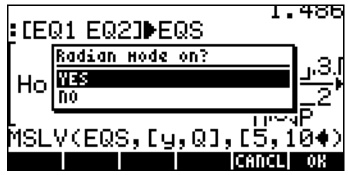
[→] [,] [←] [[]] [ALPHA] [←] [Y] [→] [,] [ALPHA] [Q] [▶]

und unseren anfänglichen Schätzungen [→] [,] [←] [[]] [5] [→] [,] [1] [0] ein.

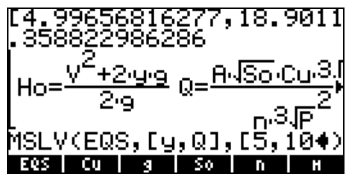
Vor drücken der Taste [ENTER], sieht die Anzeige wie folgt aus:



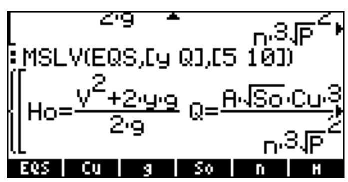
Drücken Sie [ENTER], um das System von Gleichungen zu lösen. Wenn Sie das Winkelmaß nicht auf Radian gestellt haben, könnten Sie folgende Anfrage erhalten:



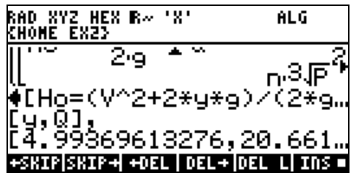
Drücken Sie OK und fahren Sie mit der Lösung fort. Ein Zwischenergebnis könnte wie folgt aussehen:



Der Vektor im oberen Teil, welcher, während der Lösungsprozess fortschreitet, die aktuellen Werte von [y,Q], und den Wert ,358822986286, welches die Kovergenz-Kriterien der numerischen Methode, die für die Lösungsfindung benutzt wurde, darstellt. Wenn das System gut eingestellt ist, wird sich dieser Wert bis ganz Nahe zu Null hin verringern. An dieser Stelle sollte eine numerische Lösung gefunden worden sein. Nachdem MSLV eine Lösung gefunden hat, sollte die Anzeige wie folgt aussehen:



Das Ergebnis ist eine Liste von drei Vektoren. Der erste Vektor stellt die gelösten Gleichungen dar. Der zweite Vektor ist die Liste der Unbekannten. Der dritte Vektor stellt die Lösung dar. Um sich diese Vektoren anzusehen, drücken Sie die Pfeiltaste ▼ zum starten des Zeileneditors. Das Ergebnis wird wie folgt angezeigt:



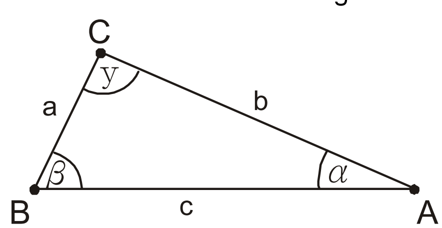
Die vorgeschlagene Lösung ist [4,99369613276,20,661… Das bedeutet, y = 4,99 ft und Q = 20,66 ft^3/s. Um die Lösung im Detail anzusehen, benutzen Sie die Pfeiltasten (◀▶▲▼).

Der Multiple Equation Solver (MES) (Mehrfachgleichungslöser)

Der Mehrfachgleichungslöser ist eine Umgebung, in der Systeme von Mehrfachgleichungen, durch lösen jeweils einer Unbekannten aus einer Gleichung, gelöst werden können. Es ist eigentlich nicht ein Löser für Simultanlösungen, sondern eher ein "eins-nach-dem-anderen" Löser für eine Anzahl von ähnlichen Gleichungen. Um die Anwendung des MES , bei der Lösung von Mehrfachgleichungen zu veranschaulichen, stellen wir im nächsten Abschnitt eine trigonometrische Anwendung vor. Die nachfolgenden Beispiele werden im RPN-Modus erzeugt:

Anwendung 1 – Lösung von Dreiecken

In diesem Abschnitt verwenden wir eine wichtige Anwendung von trigonometrischen Funktionen: die Berechnung der Maße eines Dreieckes. Die Lösung ist im Rechner, unter Verwendung des Mehrfachgleichungslösers oder kurz MES, implementiert.
Nehmen wir das Dreieck ABC aus der Abbildung unten.



Die Summe der Innenwinkel eines Dreiecks ist immer 180°, d.h. $\alpha + \beta + \gamma =$ 180°. Der Sinussatz besagt dass:

$$\frac{\sin\alpha}{a} = \frac{\sin\beta}{b} = \frac{\sin\gamma}{c}.$$

Der Cosinussatz besagt dass:

$$a^2 = b^2 + c^2 - 2 \cdot b \cdot c \cdot \cos\alpha,$$
$$b^2 = a^2 + c^2 - 2 \cdot a \cdot c \cdot \cos\beta,$$
$$c^2 = a^2 + b^2 - 2 \cdot a \cdot b \cdot \cos\gamma.$$

Um ein Dreieck lösen zu können, müssen Sie mindestens 3 der folgenden sechs Variablen kennen: *a, b, c,* α, β, γ. Dann können Sie die Gleichungen des Sinussatzes, des Cosinussatzes und der Summer der Innenwinkel anwenden, um die anderen drei Variablen zu berechnen.

Sind drei Seiten bekannt, kann die Fläche des Dreiecks mit der Heronschen Formel $A = \sqrt{s \cdot (s-a) \cdot (s-b) \cdot (s-c)}$, berechnet werden, wobei s als der Halbumfang des Dreiecks bekannt ist, d.h. $s = \frac{a+b+c}{2}$.

Lösung eines Dreiecks anhand des Multiple Equation Solvers (MES) (Mehrfachgleichungslösers)

Der Mehrfachgleichungslöser (MES) ist ein Merkmal des Rechners, das zur Lösung zweier oder mehrerer gekoppelter Gleichungen verwendet wird. Es muss noch darauf hingewiesen werden, dass der MES die Gleichungen nicht simultan löst. Die Vorgehensweise ist die, dass er die bekannten Variablen nimmt, und dann in einer Liste von Gleichungen sucht, bis eine gefunden wird, die für die unbekannten Variablen gelöst werden kann. Dann wird nach einer weiteren Gleichung gesucht, die für die nächsten Unbekannten gelöst werden kann, und so lange wiederholt, bis alle Unbekannten gelöst sind.

Erstellen eines Arbeitsverzeichnisses

Wir werden den MES zur Lösung von Dreiecken verwenden, anhand einer Liste von Gleichungen, die dem Sinus- und Cosinus-Satz, der Regel der Summe der Innenwinkel eines Dreiecks und der Heronschen Formel für die Fläche entsprechen, erstellen. Erstellen Sie als erstes im HOME Verzeichnis ein Unterverzeichnis mit dem Namen TRIANG und wechseln in dieses Verzeichnis. Siehe Kapitel 2 für Anweisungen zur Erstellung von Unterverzeichnissen.

Eingabe der Liste von Gleichungen

Geben Sie im Verzeichnis TRIANG die Liste der Gleichungen, entweder durch Direkteingabe in den Stack oder mit Hilfe des EquationWriters ein. (Beachten Sie dabei, dass ALPHA ↱ A das Zeichen α und ALPHA ↱ B das Zeichen β erzeugt. Das Zeichen γ muss über ↱ CHARS mit ECHO erzeugt werden):

‘SIN(α)/a = SIN(β)/b’
‘SIN(α)/a = SIN(γ)/c’
‘SIN(β)/b = SIN(γ)/c’
‘c^2 = a^2+b^2-2*a*b*COS(γ)’
‘b^2 = a^2+c^2-2*a*b*COS(β)’
‘a^2 = b^2+c^2-2*b*c*COS(α)’
‘α+β+γ = 180’
‘s = (a+b+c)/2’
‘A = √ (s*(s-a)*(s-b)*(s-c))’

Geben Sie anschließend die Zahl 9 ein und erstellen Sie eine Liste von Gleichungen mit der Funktion →LIST (verwenden Sie dazu den Befehle Katalog ↱ CAT). Speichern Sie die Liste in der Variablen EQ.

Die Variable EQ enthält nun eine Liste von Gleichungen, welche während des Lösungsvorgangs mit dem MES zur Lösungsfindung gescannt werden.

Titel für ein Fenster eingeben

Als nächstes erstellen wir eine String-Variable die wir TITLE benennen die den String zur "Lösung des Dreiecks" enthält, wie folgt:

↱ "" Öffnen Sie die Anführungszeichen im Stack

ALPHA ALPHA ← ALPHA	Sperrt die Tastatur für die Eingabe von Kleinbuchstaben.
← T R I A N G L E SPC	Geben Sie den Text ein: Triangle_ (Dreieck_)
← S O L U T I O N	Geben Sie den Text ein: Lösung
ENTER	Geben Sie den String "Triangle Solution" (Lösung des Dreiecks) in den Stack ein
'	Öffnen Sie die einfachen Anführungszeichen im Stack
ALPHA ALPHA T I T L E ENTER	Geben Sie den Variablennamen 'TITLE' (Titel) ein
STO▸	Speichern Sie den String in 'TITLE'

Erstellen einer Liste von Variablen

Erstellen Sie als nächstes eine Liste von Variablennamen im Stack, wie nachfolgend gezeigt:

{ a b c α β γ A s }

und speichern Sie diese in der Variablen LVARI (Liste von VARIablen). Die Liste von Variablen stellt die Reihenfolge, in der die Variablen im MES aufgelistet werden, dar, sobald dieser gestartet wird. In dieser Liste müssen alle Variablen für die Gleichungen enthalten sein oder die Funktion MTM (siehe unten) wird nicht damit funktionieren. Nachfolgend ist die Reihenfolge der Tastenanschläge, die dazu notwendig sind, die Liste zu erstellen und speichern:

Drücken Sie VAR, falls nötig, um ins Variablenmenü zu gelangen. In Ihrem Menü sollten die Variablen LVARI TITLE EQ angezeigt werden.

Vorbereitungen den MES auszuführen

Der nächste Schritt ist den MES zu starten und eine Beispiellösung zu versuchen. Bevor wir aber dies tun, setzen wir die Winkelmaße auf DEGrees (Grade), falls diese noch nicht auf Grad umgestellt sind; dazu geben wir ein: ALPHA ALPHA D E G ENTER .

Weiterhin wollen wir die Inhalte der Variablen TITLE und LVARI im Stack behalten; dies geschieht durch Verwendung von:

TITLE LVARI

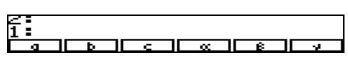
Wir werden die nachfolgenden MES-Funktionen verwenden

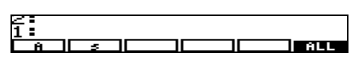
- MINIT: MES INITialization: (Initialisierung des MES) initialisiert die, in EQ gespeicherten, Variablen der Gleichungen
- MITM: MES' Menu Item: (Menüpunkt) Nimmt einen "title" aus Stack Ebene 2 und die Liste der Variablen aus Stack Ebene 1 und setzt diese über das MES Fenster und die Liste der Variablen als Funktionstasten, in der in der Liste angegebenen Reihenfolge. Im vorliegenden Beispiel haben wir bereits einen title ("Triangle Solution") und eine Variablen-Liste ({ a b c α β γ A s }) entsprechend in Stack Ebene 2 und 1 bereit zum starten des MITM.
- MSOLV: MES SOLVEr (MES Löser), startet den Mehrfachgleichungslöser (MES) und wartet auf eine Benutzereingabe.

MES interaktiv benutzen

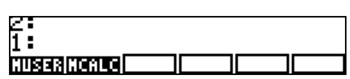
Um den MES, mit den im Stack aufgeführten Variablen LVARI und TITLE, zu starten, aktivieren Sie den Befehl MINIT, anschließend MITM und letztendlich MSOLV (diese Funktionen sind im Befehle Katalog ⮣ CAT zu finden).

Der MES wird mit der folgenden vorhandenen Liste von Variablen gestartet (Drücken Sie NXT, um die komplette Liste anzusehen):





Drücken Sie NXT ein weiteres Mal, um auch die dritte Variablenliste anzuzeigen. Sie sollten folgendes sehen:



Drücken Sie NXT noch einmal, um die erste Menüseite von Variablen anzuzeigen.

Versuchen wir eine einfache Lösung des Falles I unter Verwendung von a = 5, b = 3, c = 5. Benutzen Sie dazu folgende Einträge:

5 [a] a:5 wird in der oberen linken Ecke des Displays angezeigt.
3 [b] b:3 wird in der oberen linken Ecke des Displays angezeigt.
5 [c] c:5 wird in der oberen linken Ecke des Displays angezeigt.

Um die Winkel zu ermitteln, verwenden Sie:

↰ [α] Der Rechner meldet Solving for α, (löse für α) und zeigt das Ergebnis α: 72.5423968763.

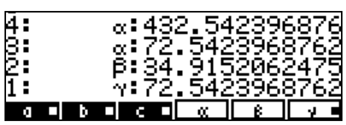
> **Anmerkung**: Ist der Wert, den Sie erhalten größer als 180, versuchen Sie folgend
>
> 1 0 [α] Re-initialize a to a smaller value.
> ↰ [α] Calculator reports Solving for α

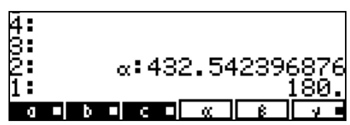
..

Anschließend berechnen wir die beiden anderen Werte:

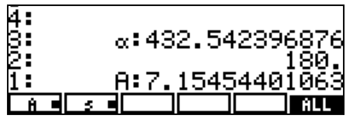
↰ [β] Die Lösung ist β: 34.9152062474.
↰ [γ] Die Lösung ist γ: 72.5423968763.

Sie sollten nun die Werte der drei aufgeführten Winkel in Stack Ebene 3 bis 1 sehen. Drücken Sie + zweimal, um zu überprüfen, ob die Summe aller drei zusammen tatsächlich 180° beträgt.



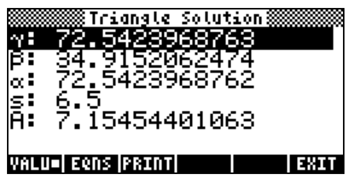


Drücken Sie die Taste NXT, um auf die nächste Seite des Menüs zu gelangen. Um nun die Fläche zu berechnen verwenden Sie: ↰ [A]. Der Rechner wird zuerst die Lösung aller anderen Variablen ermitteln und anschließend die Fläche A berechnen: 7.15454401063.



> Anmerkung: Sobald eine Lösung gefunden wurde, berichtet der Rechner die Bedingungen für die Lösung entweder als Null oder Sign Reversal (Vorzeichenänderung). Weitere Meldungen könnten auftauchen, sobald der Rechner Schwierigkeiten bei der Lösungsfindung begegnet.

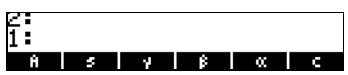
Drücken Sie nun ⬅ ALL werden alle Variablen gelöst, zeitweise werden Zwischenergebnisse angezeigt. Drücken Sie nun ➡ ALL um alle Lösungen zu sehen:



Wenn Sie damit fertig sind, drücken Sie ON, um zur MES Umgebung zurückzukehren. Drücken Sie VAR, um die MES Umgebung zu verlassen und zur Normalanzeige des Rechners zurückzukehren.

Variablen in Unterverzeichnissen organisieren

Ihr Variablen-Menü wird nun die folgenden Variablen enthalten (drücken Sie NXT, um auch die zweite Seite der Variablen anzuzeigen):



Es wurden Variablen, die allen Variablen in den Gleichungen aus EQ entsprechen, erstellt. Nun gibt es da noch eine neue Variable *Mpar* (MES Parameter), welche Informationen zu den aktuellen Einstellungen des MES für diesen einen Satz von Gleichungen enthält. Mit Hilfe von ➡ Mpar können Sie sich den Inhalt der Variablen *Mpar* anzeigen lassen. Sie erhalten folgendes kryptische Ergebnis: Library Data. (Daten zur Bibliothek) Dies bedeutet, dass die MES Parameter in einer Binärdatei kodiert sind und der Anwender auf diese nicht zugreifen kann.

Als nächstes wollen wir die Reihenfolge dieser im Menü ändern, was wir über nachfolgende Schritte machen können:

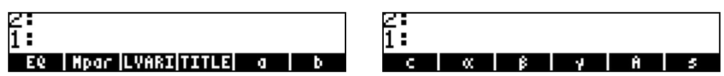
1. Erstellen Sie eine Liste, welche { EQ Mpar LVARI TITLE } enthält, unter Verwendung von:

   ← { } EQ Mpar LVARI TITLE ENTER

2. Setzen sie die Inhalte von LVARI in den Stack, unter Verwendung von: LVARI.
3. Verknüpften Sie beide Listen, indem Sie + drücken.

Verwenden Sie die Funktion ORDER (benutzen Sie dazu den Befehle Katalog → CAT), um die Reihenfolge der Variablen, die in Stack Ebene 1 angezeigt werden, zu ordnen.

4. Drücken Sie VAR um Ihre Variablen-Liste wiederherzustellen. Die Liste sollte wie folgt aussehen:



5. Drücken Sie NXT um das erste Variablen-Menü wiederherzustellen.

Programmieren der MES Dreiecks-Lösung über User RPL

Um das Starten des MES für spätere Anwendungen zu vereinfachen, erstellen wir ein Programm, das den MES mit einer einzigen Taste starten wird. Das Programm sollte wie folgt aussehen: << DEG MINIT TITLE LVARI MITM MSOLVR >> und kann wie folgt eingegeben werden:

→ « »	Öffnet das Programmsymbol
ALPHA ALPHA	Verriegelt die alphanumerische Tastatur
D E G SPC	Geben Sie DEG (Winkelmaß wird auf DEGress (Grad) gesetzt)
M I N I T SPC	Geben Sie MINIT_ ein
ALPHA	Entsperrt die alphanumerische Tastatur
TITLE	Listen Sie den Namen TITLE im Programm auf

LVARI	Listen Sie den Namen LVARI im Programm auf
ALPHA ALPHA	Verriegelt die alphanumerische Tastatur
M I T M SPC	Geben Sie MITM_ ein
M S O L V R	Geben Sie MSOLVR
ENTER	Geben Sie program in den Stack ein

Speichern Sie program in einer Variablen mit dem Namen TRISOL, für TRIangle SOLution (Lösung für das Dreieck) unter Verwendung von:
' ALPHA ALPHA T R I S O L ENTER STO▸

Drücken Sie VAR, falls nötig, um Ihre Variablenliste wieder herzustellen. In Ihrem Menü sollte jetzt die Funktionstaste TRISO zur Verfügung stehen.

Das Programm ausführen - Lösungsbeispiele

Um das Programm auszuführen, drücken Sie die Funktionstaste TRISO . Sie erhalten nun das MES Menü, das der Lösung des Dreiecks entspricht. Versuchen wir nun einige Beispiele von drei, weiter oben aufgeführten Fällen, für die Lösung des Dreiecks.

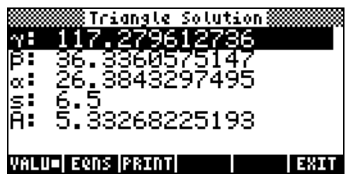
Beispiel 1 – rechtwinkliges Dreieck
Verwenden Sie a = 3, b = 4, c = 5. Hier ist die Lösungsfolge:

3 [a] 4 [b] 5 [c]	um Daten einzugeben
↰ [α]	Die Lösung ist α: 36.8698976458
↰ [β]	Die Lösung ist β: 53.1301023541.
↰ [γ]	Die Lösung ist γ: 90.
NXT	Um zum nächsten Variablenmenü zu gelangen.
[↰][A]	Die Lösung ist A: 6.
NXT NXT	Um zum nächsten Variablenmenü zu gelangen.

Beispiel 2 – Irgend ein Dreieck
Verwenden Sie a = 3, b = 4, c = 6. Die hier verwendete Lösungsprozedur besteht darin, alle Variablen gleichzeitig zu lösen und anschließend die Lösung in den Stack zu laden.

VAR TRISO	Um die Daten zu löschen und den MES neu zu starten
3 [a] 4 [b] 6 [c]	um Daten einzugeben
NXT	Um zum nächsten Variablenmenü zu gelangen.
← ALL	Lösen aller Unbekannten.
→ ALL	Zeige die Lösung:

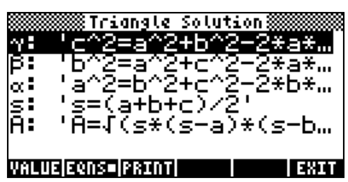
Die Lösung lautet:



Am unteren Rand der Anzeige haben Sie die Funktionstasten:

VALU▪ EQNS PRINT EXIT

Das kleine schwarze Rechteck neben dem Namen der Funktion VALU▪ zeigt an, dass die Werte und nicht die Gleichungen aus denen diese berechnet wurden, angezeigt werden. Um sich die für die Lösung verwendeten Gleichungen anzusehen, drücken Sie die Funktionstaste EQNS. Die Anzeige sieht wie folgt aus:



Die Funktionstaste PRINT wird dazu benutzt den Bildschirminhalt auf einem Drucker, falls vorhanden, auszugeben. EXIT bringt Sie zur MES Umgebung zurück, um eine neue Lösung, falls benötigt, zu ermitteln. Um zur Normalanzeige zurückzukehren, drücken Sie VAR.

Die nachfolgende Tabelle von Lösungen für Dreiecke zeigt die Dateneingabe fettgedruckt und die Lösungen in Kursivschrift. Um die Lösungen zu überprüfen, versuchen Sie das Programm mit diesen Eingaben auszuführen. Vergessen Sie nicht VAR TRISO am Ende jeder Lösung einzugeben, um die Variableninhalte zu löschen und starten Sie die MES Lösung erneut. Andernfalls könnten Sie Informationen aus vorangegangenen Lösungen noch mit sich ziehen, welches Ihre derzeitigen Berechnungen durcheinanderbringen könnte.

a	***b***	***c***	$\alpha(^o)$	$\beta(^o)$	$\gamma(^o)$	***A***
2.5	*6.9837*	**7.2**	*20.299*	**75**	*84.771*	*8.6933*
7.2	**8.5**	*14.26*	*22.616*	**27**	*130.38*	*23.309*
21.92	**17.5**	**13.2**	**90**	*52.97*	**37.03**	*115.5*
41.92	**23**	*29.6*	**75**	**32**	*73*	*328.81*
10.27	*3.26*	**10.5**	*77*	**18**	**85**	*16.66*
17	**25**	**32**	*31.79*	*50.78*	*97.44*	*210.71*

Hinzufügen einer INFO Schaltfläche zu Ihrem Verzeichnis

Eine Informationsschaltfläche hilft Ihnen, sich die Operationen und Funktionen des Verzeichnisses zu merken. Das einzige, was wir in diesem Verzeichnis wissen sollten, ist, dass man durch drücken von TRISO die Lösung des Dreiecks startet. Sie sollten folgendes Programm eingeben: <<"Drücken Sie [TRISO] um zu starten." MSGBOX >> und speichern Sie dies in einer Variablen INFO. Als Ergebnis, wird die erste Variable in Ihrem Verzeichnis nun die INFO Schaltfläche sein.

Anwendung 2 – Geschwindigkeit und Beschleunigung in Polarkoordinaten

Zweidimensionale Partikelbewegung in Polarkoordinaten beinhaltet oft die Ermittlung der Radial- und Transversal-Komponenten der Geschwindigkeit und Beschleunigung der gegebenen Partikel r, $r' = dr/dt$, $r'' = d^2r/dt^2$, θ, $\theta' = d\theta/dt$ und, $\theta'' = d^2\theta/dt^2$.. Nachfolgende Gleichungen werden verwendet:

$$v_r = \dot{r} \qquad a_r = \ddot{r} - r\dot{\theta}^2$$
$$v_\theta = r\dot{\theta} \qquad a_\theta = r\ddot{\theta} + 2\dot{r}\dot{\theta}$$

Erstellen Sie ein Unterverzeichnis POLC (POLar Coordinates – Polarkoordinaten), welches wir bei der Berechnung der Geschwindigkeit und Beschleunigung in Polarkoordinaten verwenden. Geben Sie die nachfolgenden Variablen in dieses Unterverzeichnis ein:

Programm oder Wert	zu speichern in Variable:
<< PEQ STEQ MINIT NAME LIST MITM MSOLVR >>	**SOLVEP**
"vel. & acc. polar coord."	**NAME**
{ r rD rDD θD θDD vr vθ v ar aθ a }	**LIST**
{ 'vr = rD' 'vθ = r*θD' 'v = √(vr^2 + vθ^2)' 'ar = rDD – r*θD^2' 'aθ = r*θDD + 2*rD*θD' 'a = √(ar^2 + aθ^2)' }	**PEQ**

Nachfolgend Erläuterungen zu den Variablen:

SOLVEP = ein Programm, welches den Mehrfachgleichungslöser, für die in der Variablen **PEQ** gespeicherten Gleichungen, startet

NAME = eine Variable, in welcher der Name des Mehrfachgleichungslösers, und zwar*, "vel. & acc. polar coord."* speichert

LIST = eine Liste, der in den Berechnungen, verwendeten Variablen, in der Reihenfolge eingegeben, in der wir diese in der Mehrfachgleichungslöser-Umgebung anzeigen lassen möchten

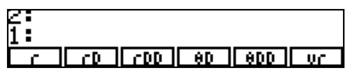
PEQ = Liste von Gleichungen die gelöst werden sollen, entsprechend der Radial- und Transversal-Komponenten der Geschwindigkeit (**vr**, **v**θ) und der Beschleunigung (**ar**, **a?**) in Polarkoordinaten, sowie die Gleichungen zur Berechnung der Magnitude der Geschwindigkeit (**v**) und der Beschleunigung (**a**), wenn die Polarkomponenten bekannt sind.

r, **rD**, **rDD** = r (Radialkoordinaten), r-dot (erste Ableitungsfunktion von r), r-double dot (zweite Ableitungsfunktion von r).

θ**D**, θ**DD** = θ-dot (erste Ableitungsfunktion von θ), θ-double dot (zweite Ableitungsfunktion von θ).

Angenommen folgende Informationen sind Ihnen bekannt: r = 2,5, rD = 0,5, rDD = -1,5, θD = 2,3, θDD = -6,5 und Sie müssen die Werte für vr, vθ, ar, aθ, v und a ermitteln.

Starten Sie den Merfachgleichungslöser, durch drücken von VAR SOLVE . Im Display erscheint eine Anzeige mit der Beschriftung , *"vel. & acc. polar coord."*, die wie folgt aussieht:



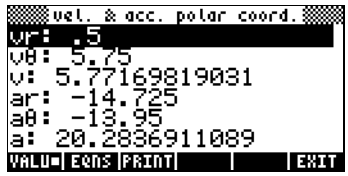
Um die Werte der bekannten Variablen einzutragen, geben Sie einfach den Wert ein und drücken Sie die Taste, die dieser Variablen entspricht. Verwenden Sie dazu folgende Tastenfolge: 2,5 [r] 0,5 [rD] 1,5 +/- [rDD] 2,3 [θD] 6,5 +/- [θDD].

Beachten Sie dabei, dass wenn Sie einen bestimmten Wert eingegeben haben, erscheint die Variable mit dem entsprechenden Wert in der linken oberen Ecke des Displays. Nun haben wir die bekannten Variablen eingegeben. Die Ermittlung der Unbekannten kann auf zwei verschiedene Arten erfolgen:

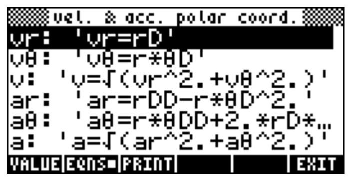
a). Lösen Sie für einzelne Variablen, z.B. , ↰[vr] ergibt vr: 0,500. Drücken Sie NXT ↰[vθ] um vθ : 5,750 zu erhalten und so weiter. Die noch verbleibenden Ergebnisse sind - v: 5,77169819031; *a*r: -14,725; *a*θ: -13,95 und *a*: 20,2836911089.; oder

b). Lösen Sie alle Variablen auf einmal, indem Sie ↰ ALL drücken. Im Display werden die Werte, sobald diese ermittelt wurden, angezeigt.

Sobald die Berechnung beendet ist, können Sie ↱ ALL drücken, um alle Ergebnisse anzuzeigen. In diesem Fall haben wir:

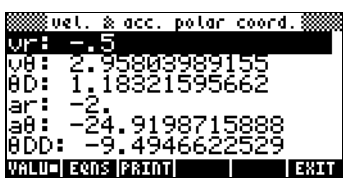


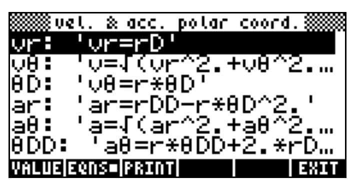
Drücken Sie die Funktionstaste EQNS, erhalten Sie die Gleichungen, für jeden einzelnen Wert in der Anzeige, die zur Lösung benutzt wurden:



Um einen neuen Satz von Werten zu verwenden, drücken Sie entweder EXIT ALL NXT NXT oder VAR SOLVE.

Versuchen wir in weiteres Beispiel mit den Werten - r = 2,5, vr = rD = -0,5, rDD = 1,5, v = 3,0, a = 25,0. Ermitteln Sie θD, θDD, vθ, ar und aθ. Sie sollten folgendes Ergebnis erhalten:





Kapitel 8
Operationen mit Listen

Listen sind Objekte des Rechners, die besonders bei der Datenverarbeitung und in der Programmierung hilfreich sein können. In diesem Kapitel werden Beispiele von Operation mit Listen vorgestellt.

Definitionen

Im Kontext des Rechners wird eine Liste, eine Reihe von Objekten, eingeschlossen in ein Klammerpaar, getrennt durch Leerschritte (SPC) im RPN-Modus oder Kommas (→ ,) in beiden Modi, bezeichnet. Objekte die in eine Liste eingefügt werden können sind: Zahlen, Buchstaben, Strings, Namen von Variablen und/oder Operatoren. Listen sind besonders bei der Manipulation von Daten-Sets und in einigen Programmieranwendungen von Bedeutung. Beispiele von Listen sind:

```
{ t 1 }, {"BETA" h2 4},{1 1.5 2.0},
{a a a a},{ {1 2 3} {3 2 1} {1 2 3}}
```

In den unten aufgeführten Beispielen beschränken wir uns auf numerische Listen.

Erstellen und speichern von Listen

Um eine Liste im ALG-Modus zu erstellen, drücken Sie als erstes die Klammertaste ← {} (der Taste + zugeordnet), geben Sie anschließend die Elemente der Liste ein, und trennen Sie diese durch ein Komma voneinander (→ ,). Mit den nachfolgenden Tastenanschlägen geben Sie die Liste {1 2 3 4} ein und speichern diese in der Variablen L1.

← {} 1 → , 2 → , 3 → , 4
▶ STO▶ ALPHA L 1 ENTER

In der Anzeige erscheint folgendes:

Die linke Abbildung zeigt die Anzeige vor drücken der Taste [ENTER], während auf der rechten Seite die Anzeige, nach speichern der Liste in L1 angezeigt wird. Beachten Sie, vor drücken der Taste [ENTER], dass in der Liste die Elemente durch ein Komma getrennt dargestellt werden. Nachdem Sie nun die Taste [ENTER] drücken, verschwinden die Kommas und die Elemente sind durch Leerschritte voneinander getrennt.

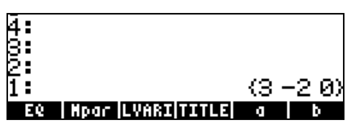
Um dieselbe Liste im RPN-Modus einzugeben, verwenden Sie die Tastenfolge:

[←]{ } [1] [SPC] [2] [SPC] [3] [SPC] [4] [ENTER]
[ALPHA][L][1][ENTER][▶][STO▶]

Die Abbildung zeigt den RPN Stack vor drücken der Taste [STO▶]:

Erstellen und Zerlegen von Listen

Erstellen und Zerlegen von Listen macht nur im RPN-Modus Sinn. In diesem Operationsmodus, benutzen wir die Funktion OBJ→ zum zerlegen der Liste. Mit dieser Funktion, wird im RPN Stack eine Liste in ihre Elemente zerlegt, wobei in Stack Ebene 1: die Anzahl der Elemente in der Liste angezeigt wird. Die nächsten beiden Abbildungen zeigen den Stack mit einer kleinen Liste vor und nach Anwendung der Funktion OBJ→:

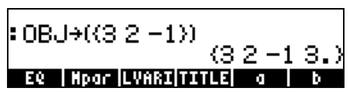


Beachten Sie, dass nach Anwenden der Funktion OBJ→ die Elemente der Liste in den Ebenen 4: bis 2: angezeigt werden, während Ebene 1: die Anzahl der Elemente in der Liste enthält.

Um im RPN-Modus eine Liste zu erstellen, geben Sie erst die Elemente in den Stack ein, dann Größe der Liste und starten Sie die Funktion →LIST (wählen

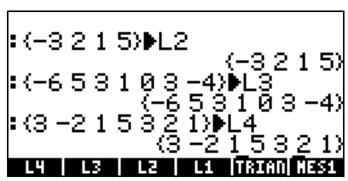
Sie diese aus dem Katalog mit folgender Tastenkombination:
↱ _CAT_ ↱ →, anschließend suchen Sie diese mit den Pfeiltasten (▲▼) und wählen diese aus). In den nachfolgenden Abbildungen sehen Sie eine Liste der Größe 4, vor und nach Anwender der Funktion →LIST:

Anmerkung: Die Funktion OBJ→, wenn im ALG-Modus angewandt, wiedergibt einfach die Liste und fügt dieser die Listengröße hinzu:



Operationen mit Zahlenlisten

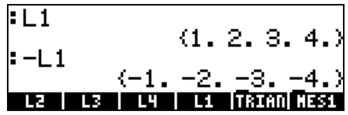
Um Operationen mit Zahlenlisten zu veranschaulichen, werden wir, zusätzlich zu der oben erstellten L1, einige weitere Listen erzeugen: L2={-3,2,1,5}, L3={-6,5,3,1,0,3,-4}, L4={3,-2,1,5,3,2,1}. Im ALG-Modus sieht die Anzeige, nachdem Sie die Listen L2, L3 und L4 eingegeben haben, wie folgt aus:



Im RPN-Modus werden drei Listen und deren Namen, bereit zum speichern, angezeigt. Um die Listen in diesem Fall zu speichern, drücken Sie dreimal die Taste STO▶.

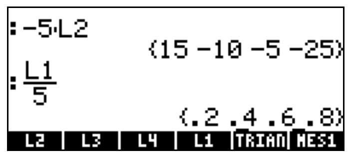
Änderung des Vorzeichens

Wenn auf eine Liste von Zahlen angewandt, ändert die Taste "Vorzeichen ändern" (+/-) das Vorzeichen aller Elemente in der Liste. So zum Beispiel:

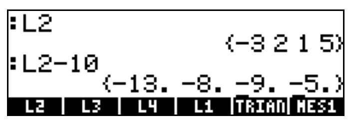


Addition, Subtraktion, Multiplikation, Division

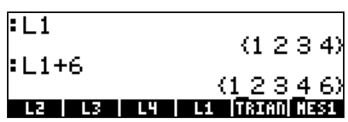
Multiplikation und Division einer Lise durch eine einzelne Zahl, wird über die gesamte Liste angewandt, so z.B.:



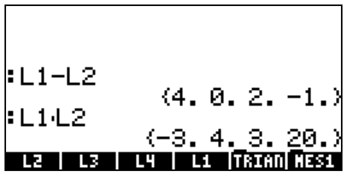
Subtraktion einer einzelnen Zahl von einer Liste, wird die Zahl von jedem Element der Liste abziehen, so z.B.:

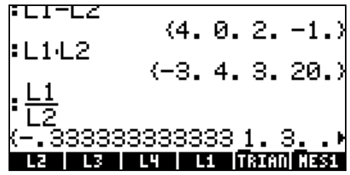


Addition einer einzelnen Zahl zu einer Liste, erzeugt eine um diese Zahl erhöhte Liste, aber nicht eine Addition zu jedem einzelnen Element der Liste. So zum Beispiel:

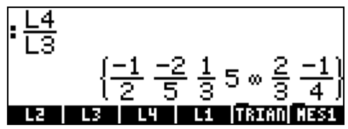


Subtraktion, Multiplikation und Division von Listen mit einer Zahlenreihe der gleichen Länge, erzeugt eine Liste der gleichen Länge mit Glied für Glied Operationen. Beispiele:



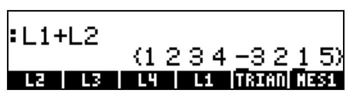


Die Division L4/L3 ergibt einen unendlichen Eintrag, weil eines der Elemente in L3 eine Null ist:

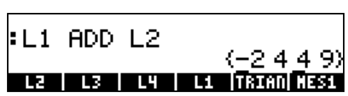


Haben die Listen für Rechenoperation verschiedene Längen, wird eine Fehlermeldung (Error: Invalid Dimensions – Fehler: ungültige Dimensionen) ausgegeben.

Wird das Pluszeichen (+) auf Listen angewandt, verhält sich dieses als *Verkettungsoperator*, indem es zwei Listen zusammenfügt und nicht Glied für Glied addiert. So zum Beispiel:



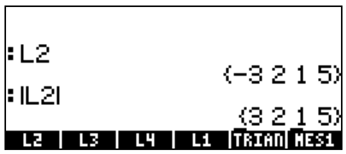
Um eine Glied für Glied Addition zweier Listen gleicher Länge zu erzielen, muss der Operator ADD verwendet werden. Dieser Operator kann mit Hilfe der Funktion Katalog (→ CAT) geladen werden. In der nachfolgenden Abbildung wird eine ADD Anwendung gezeigt, um die Listen L1 und L2, Glied für Glied zu addieren:



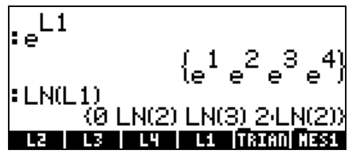
Funktionen mit reellen Zahlen von der Tastatur aus

In Listen können auch Funktionen mit reellen Zahlen (ABS, e^x, LN, 10^x, LOG, SIN, x^2, √, COS, TAN, ASIN, ACOS, ATAN, y^x) von der Tastatur aus verwendet werden. Hier einige Beispiele:

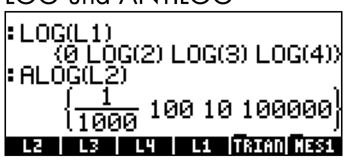
ABS



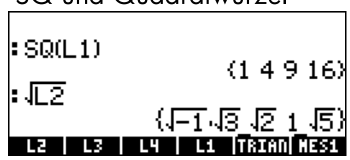
EXP und LN



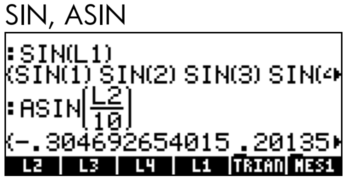
LOG und ANTILOG



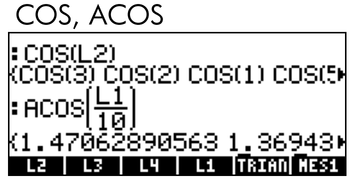
SQ und Quadratwurzel



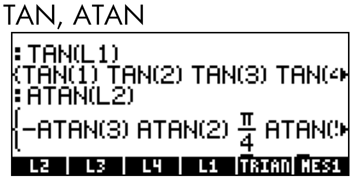
SIN, ASIN



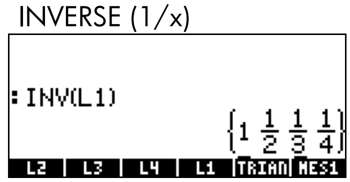
COS, ACOS



TAN, ATAN



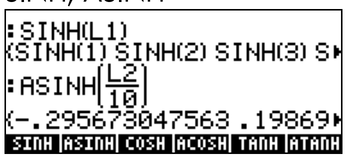
INVERSE (1/x)



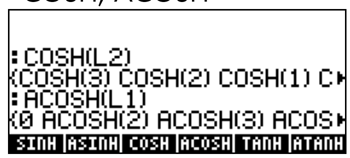
Funktionen von reellen Zahlen aus dem Menü MTH

Funktionen des MTH-Menüs die von Interesse sind - aus dem Menü HYPERBOLIC: SINH, ASINH, COSH, ACOSH, TANH, ATANH und aus dem Menü REAL: %, %CH, %T, MIN, MAX, MOD, SIGN, MANT, XPON, IP, FP, RND, TRNC, FLOOR, CEIL, D→R, R→D. Nachfolgend einige der Funktionen die nur ein Argument benötigen und auf Listen von reellen Zahlen angewendet werden können:

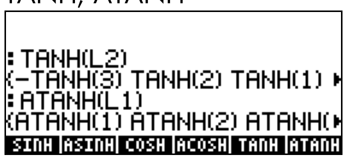
SINH, ASINH



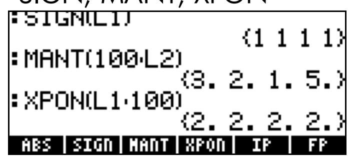
COSH, ACOSH



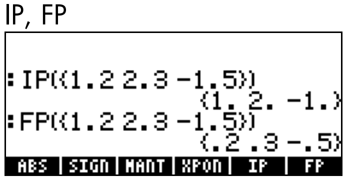
TANH, ATANH



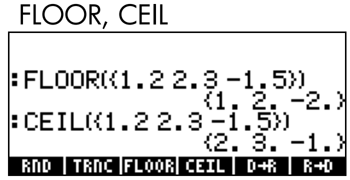
SIGN, MANT, XPON



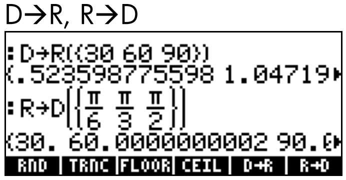
IP, FP



FLOOR, CEIL

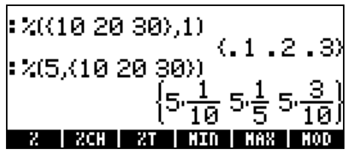


D→R, R→D



Beispiele von Funktionen mit zwei Argumenten

In den nachfolgenden Abbildungen finden Sie Anwendungen der Funktion % zur Auflistung von Argumenten. Die Funktion % benötigt zwei Argumente. Das erste der beiden Beispiele zeigt Fälle, in denen nur eines der beiden Argumente eine Liste darstellt.



Das Ergebnis sind Listen mit der Funktion % aufgeteilt gemäß der Liste der Argumente. So zum Beispiel:

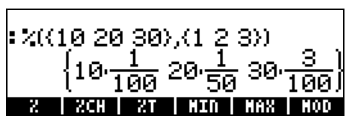
%({10, 20, 30},1) = {%(10,1),%(20,1),%(30,1)},

während

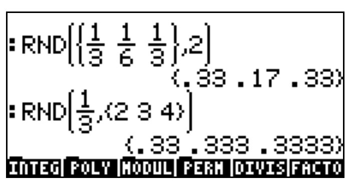
%(5,{10,20,30}) = {%(5,10),%(5,20),%(5,30)}

Im nachfolgenden Beispiel, sind beide Argumente der Funktion % Listen derselben Größe. In diesem Fall, wird eine Glied-für-Glied Verteilung der Argumente durchgeführt, d.h.

%({10,20,30},{1,2,3}) = {%(10,1),%(20,2),%(30,3)}

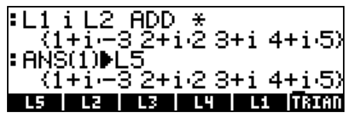


Diese Beschreibung der Funktion % für die Auflistung von Argumenten, zeigt ein allgemeines Muster für die Auswertung jeder Funktion mit zwei Argumenten, wobei entweder eines der Argumente oder beide Listen sind. Nachfolgend einige Anwendungsbeispiele der Funktion RND.

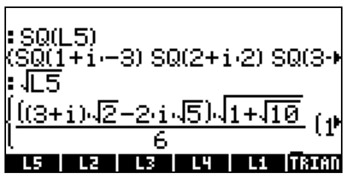


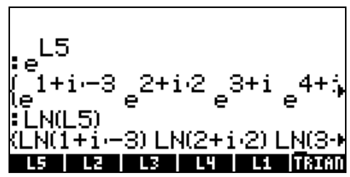
Listen von komplexen Zahlen

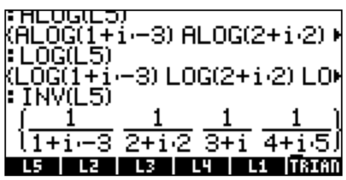
Das nachfolgende Beispiel zeigt wie man eine Liste von komplexen Zahlen erstellt, vorausgesetzt, beide Listen sind gleich lang, wobei eine Liste die reellen Teile während die zweite die imaginären Teile der komplexen Zahlen, darstellt. Verwenden Sie L1 ADD i*L2. Die Anzeige zeigt auch, dass die daraus resultierende komplexe Zahlenliste in der Variablen L5 gespeichert wird:

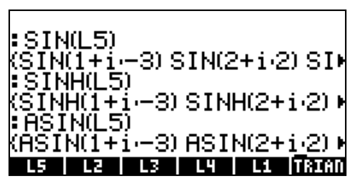


Auch Funktionen wie LN, EXP, SQ, usw. können auf Listen von komplexen Zahlen angewandt werden, z.B.:

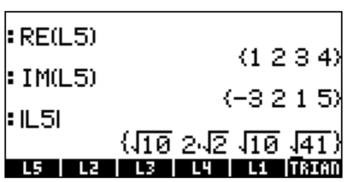


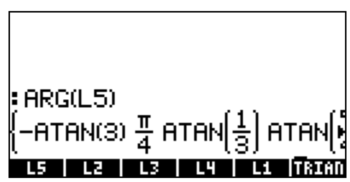






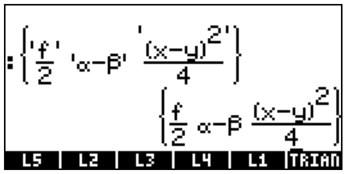
Das nachfolgende Beispiel zeigt Anwendungen der Funktion RE(reeller Teil), IM(imaginärer Teil), ABS(Magnitude) und ARG(Argument) für komplexe Zahlen. Das Ergebnis ist eine Liste von reellen Zahlen:

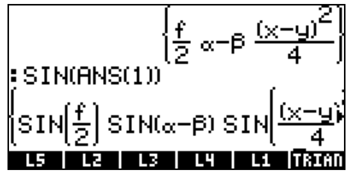




Listen von algebraischen Objekten

Nachfolgend Beispiele von Listen mit algebraischen Objekten auf welche die Funktion SIN angewendet wurde:





Das MTH/LIST Menü

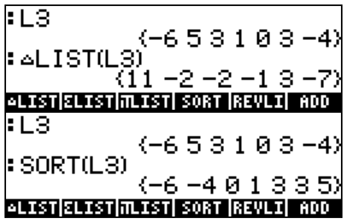
Das Menü MTH stellt eine Reihe von Funktionen, die exklusiv auf Listen angewandt werden können, zur Verfügung. Mit dem Flag 117 auf CHOOSE Kästchen gesetzt:

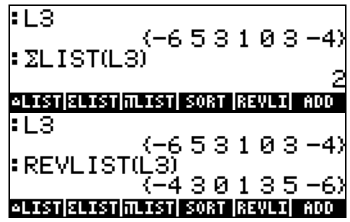
Als nächstes System Flag auf 117 auf SOFT Menüs gesetzt.

In diesem Menü finden wir die nachfolgenden Funktionen:

ΔLIST : Berechnet das Inkrement zwischen aufeinanderfolgenden Elementen in der Liste
ΣLIST : Berechnet die Summe der Elemente in der Liste
ΠLIST : Berechnet das Produkt der Elemente in der Liste
SORT : Sortiert die Elemente aufsteigend
REVLIST : Kehrt die Reihenfolge in der Liste um
ADD : Operator für die Glied-für-Glied Addition zweier Listen der gleichen Länge (Beispiele für diesen Operator wurden oben gezeigt)

Nachfolgend einige Anwendungsbeispiele dieser Funktionen im ALG-Modus.

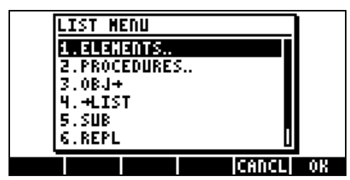




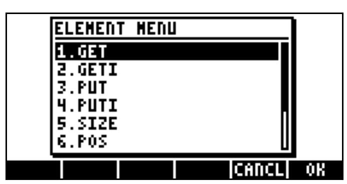
SORT und REVLIST können kombiniert werden, um eine Liste in absteigender Folge zu sortieren.

Manipulation der Elemente einer Liste

Das PRG (Programmier) Menü enthält eine Untermenü LIST mit verschiedenen Funktionen, die zur Manipulation von Elementen in einer Liste dienen.   Mit dem System Flag 117 auf CHOOSE Kästchen gesetzt:

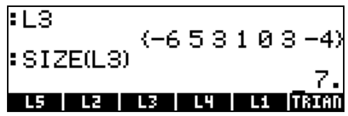


Position 1. ELEMENTS.. enthält die folgenden Funktionen, die zur Manipulation von Elementen in Listen dienen:



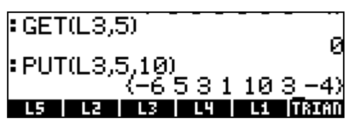
Listengröße

Die Funktion SIZE (Größe) aus dem Untermenü PRG/LIST/ELEMENTS, kann zur Ermittlung der Größe (auch als Länge bekannt) der Liste verwendet werden, so z.B.



Extrahieren und einfügen von Elementen in eine Liste

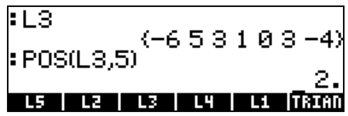
Um Elemente aus einer Liste zu extrahieren, benutzen wir die Funktion GET, welche im Untermenu PRG/LIST/ELEMENTS zu finden ist. Die Argumente der Funktion GET bilden eine Liste und die Anzahl der Elemente, die Sie aus dieser Liste extrahieren möchten. Um Elemente in eine Liste einzufügen, benutzen wir die Funktion PUT, welche ebenfalls im Untermenu PRG/LIST/ELEMENTS zu finden ist. Die Argumente der Funktion PUT bilden eine Liste, die Position welche wir ersetzen möchten und den Wert mit dem diese ersetzt werden soll. Anwendungsbeispiele für die Funktionen GET und PUT finden Sie nachfolgend:



Die Funktionen GETI und PUTI, auch im Unterverzeichnis PRG/ ELEMENTS/ zu finden, können auch dazu verwendet werden Elemente aus einer Liste zu extrahieren oder einzufügen. Diese beiden Funktion, werden aber hauptsächlich in der Programmierung verwendet. Die Funktion GETI benutzt die gleichen Argumente wie GET und gibt eine Liste, die Position der Elemente plus eine und das Element an der gewünschten Position wieder. Die Funktion PUTI verwendet die gleichen Argumente wie GET und gibt sowohl die Liste als auch die Listengröße zurück.

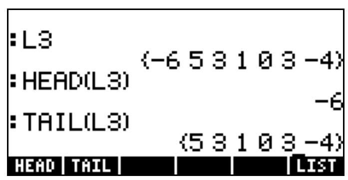
Position eines Elementes in der Liste

Zur Bestimmung der Position eines Elementes in einer Liste verwenden Sie die Funktion POS, welche die Liste und das gewünschte Element als Argument enthält. So zum Beispiel:



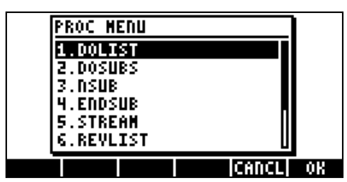
Die Funktionen HEAD und TAIL

Die Funktion HEAD extrahiert das erste Element der Liste. Die Funktion TAIL entfernt das erste Element einer Liste und gibt die noch verbleibende Liste zurück. Nachfolgend einige Beispiele:



Die Funktion SEQ

Position 2. PROCEDURES.. im Menü PRG/LIST enthält folgende Funktionen, die bei Listenoperationen Anwendung finden.

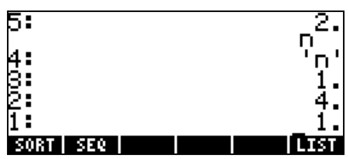


Die Funktionen REVLIST und SORT wurden bereits als Teil des Menüs MTH/LIST vorgestellt. Die Funktionen DOLIST, DOSUBS, NSUB, ENDSUB und STREAM sind zur Programmierung von Funktionen für Operationslisten im RPN-Modus gedacht. Die Funktion SEQ dient zur Erstellung einer Werteliste, bei einem bestimmten gegebenen Ausdruck, und wird nachfolgend ausführlich beschrieben.

Die Funktion SEQ enthält als Argumente einen Ausdruck in Form eines Index, den Namen dieses Index und Start- und Endwerte, sowie deren Inkrement und gibt eine Liste wieder, die aus der Auswertung des Ausdruckes für alle möglichen Werte des Index, zusammengesetzt ist. Die allgemeine Form der Funktion ist SEQ(*expression, index, start, end, increment - Ausdruck, Index, Start, Ende, Inkrement*).

Im nachfolgenden Beispiel, im ALG-Modus, stellen wir fest dass der *Ausdruck* = n^2, der *Index* = n, *Start* = 1, *Ende* = 4 und *Inkrement* = 1 ist.

Die erzeugte Liste entspricht den Werten {1^2, 2^2, 3^2, 4^2}. Im RPN-Modus können Sie unterschiedliche Argumente der Funktion wie folgt auflisten:

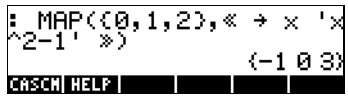


vor Anwendung der Funktion SEQ.

Die Funktion MAP

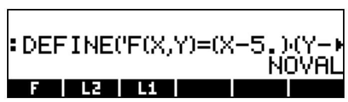
Die Funktion MAP, welche ebenfalls über den Befehle Katalog (↱ CAT), aufgerufen werden kann, nimmt als Argumente eine Liste von Zahlen und eine Funktion f(X) oder ein Programm << → a … >> und erzeugt eine Liste die aus der Anwendung der Funktion f oder des Programms auf die Zahlenliste entsteht. So z.B., wendet der nachfolgende Funktionsaufruf von MAP eine SIN(X) Funktion auf die Liste {1,2,3} an:

Der nachfolgende Aufruf der Funktion MAP verwendet als zweites Argument ein Programm anstelle einer Funktion:

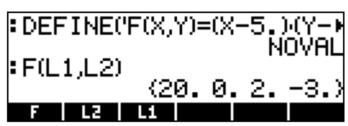


Funktionen definieren die Listen benutzen

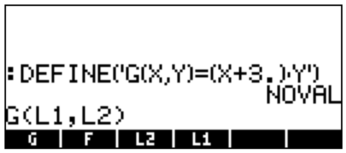
In Kapitel 3 haben wir die Funktion DEFINE (DEF) , die zum erzeugen von Funktionen aus reellen Zahlen, mit einem oder mehreren Argumenten, dient, vorgestellt. Eine über DEF definierte Funktion wird auch mit Argumentlisten verwendet, ausgenommen, dass man bei jeder Funktion die eine Addition enthält den Operator ADD und nicht das Pluszeichen (+) verwenden muss. So z.B., wenn wir im ALG-Modus (nachfolgend gezeigt) die Funktion F(X,Y) = (X-5)*(Y-2) definieren:

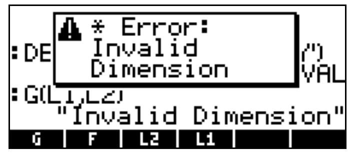


können wir Listen (z.B. die Variablen L1 und L2, welche wir weiter oben in diesem Kapitel definiert haben) zur Auswertung der Funktion verwenden, was zu folgendem Ergebnis führt:

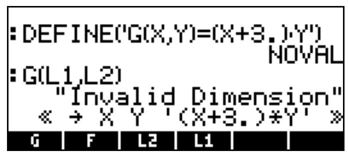


Da in der Funktion keine Addition vorkommt, ist die Anwendung dieser Funktion zur Auflistung von Argumenten recht einfach. Wenn wir aber die Funktion G(X,Y) = (X+3)*Y erstellen, wird ein Versuch diese Funktion mit Argumentlisten (L1, L2) zu berechnen, fehlschlagen:

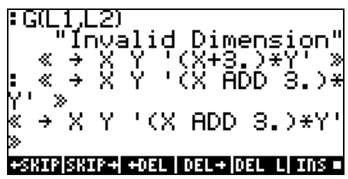




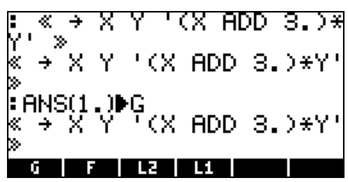
Um dieses Problem zu beheben, können wir die Inhalte der Variablen G , welche wir im Stack über die Tasten ↱ G anzeigen können, bearbeiten:



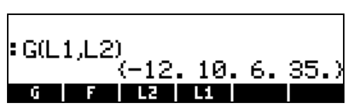
um das Pluszeichen (+) mit ADD zu ersetzen:



Anschließend speichern wir den bearbeiteten Ausdruck in der Variablen G:

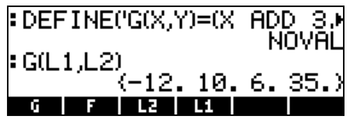


Die Auswertung G(L1,L2) ergibt nur folgendes Ergebnis:



Alternativ dazu können Sie die Funktion mit ADD anstelle des Pluszeichens (+) von Anfang an definieren, d.h. Sie verwenden

DEFINE('G(X,Y)=(X ADD 3)*Y'):



Sie können die Funktion aber auch als G(X,Y) = (X–3)*Y definieren.

Anwendungen für Listen

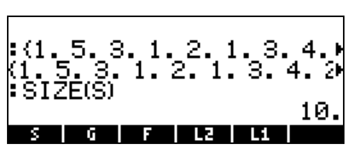
Dieser Abschnitt zeigte eine Reihe von Anwendungen von Listen, zur Berechnung von Statistiken einer Stichprobe. Unter Stichprobe verstehen wir eine Liste von Werten, sagen wir {s_1, s_2, …, s_n}. Angenommen diese Stichprobe ist die Liste

{1, 5, 3, 1, 2, 1, 3, 4, 2, 1}

und wir speichern diese in eine Variable mit dem Namen S (in der nachfolgenden Abbildung sehen Sie dies im ALG-Modus, das Erstellen dieser im RPN-Modus ist sehr ähnlich. Vergessen Sie nur nicht, dass Sie im RPN-Modus erst die Argument zu den Funktionen in den Stack eingeben müssen und erst dann die Funktion starten):

Harmonischer Mittelwert einer Liste

Dieses Muster ist klein genug, um die Anzahl der Elemente im Display abzählen zu können (n=10). Für eine größere Liste, können wir die Funktion SIZE benutzen, um die Anzahl der Elemente in der Liste anzuzeigen, z.B.

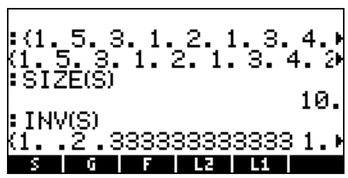


Angenommen wir wollen das harmonische Mittel der Stichprobe, welches wie nachfolgend definiert ist, berechnen

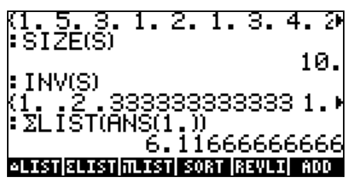
$$s_h = \frac{1}{\frac{1}{n}\sum_{k=1}^{n}\frac{1}{s_n}} = \frac{1}{\frac{1}{n}\left(\frac{1}{s_1}+\frac{1}{s_2}+\cdots+\frac{1}{s_n}\right)}.$$

Um diesen Wert zu berechnen können wir wie folgt vorgehen:

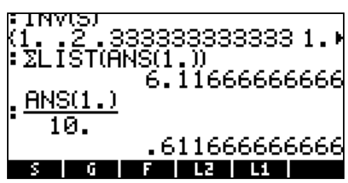
1. Wenden Sie die Funktion INV() auf die Liste S an:
2.



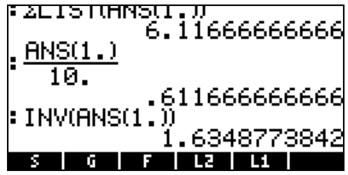
3. Wenden Sie nun auf die in Ebene 1 erhaltene Liste die Funktion ΣLIST() an.



4. Teilen Sie das obige Ergebnis durch n = 10:



5. Wenden Sie auf das letzte Ergebnis die Funktion INV() an:



Somit ist der harmonische Mittelwert einer Liste S gleich s_h = 1,6348…

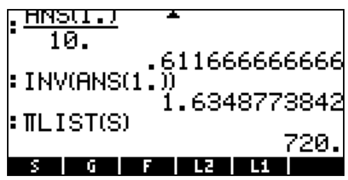
Geometrischer Mittelwert einer Liste

Der geometrische Mittelwert einer Stichprobe wird wie folgt definiert

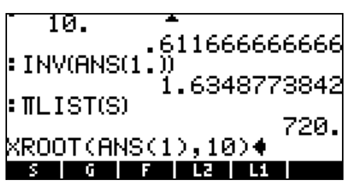
$$x_g = \sqrt[n]{\prod_{k=1}^{n} x_k} = \sqrt[n]{x_1 \cdot x_2 \cdots x_n}$$

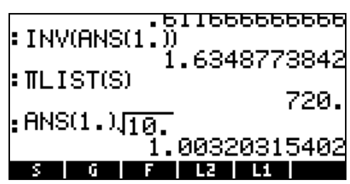
Um den geometrischen Mittelwert einer in S gespeicherten Liste zu berechnen, können wir wie folgt vorgehen:

1. Wenden Sie die Funktion ΠLIST() auf die Liste S an:



2. Wenden Sie die Funktion XROOT(x,y), d.h. die Tastenfolge ⬏ $\sqrt[x]{y}$ an, um das Ergebnis in Ebene 1 zu erhalten:





Somit ist der geometrische Mittelwert einer Liste S gleich s_h = 1,003203…

Gewogenes Mittel

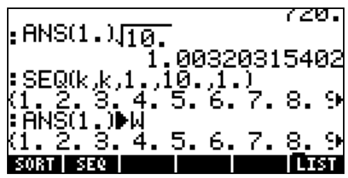
Angenommen die Daten in Liste S, wie oben definiert, d.h.:

S = {1,5,3,1,2,1,3,4,2,1}

werden von folgenden Gewichten beeinflusst

W = {1, 2, 3, 4, 5, 6, 7, 8, 9, 10}

Wenn wir die Liste der Gewichte als W = $\{w_1,w_2,\ldots,w_n\}$ erstellen, stellen wir fest, dass das k-te Element in der Liste W, durch $w_k = k$ definiert werden kann. Somit können wir die Funktion SEQ dazu verwenden eine Liste zu generieren und diese dann in die Variable wie folgt speichern:

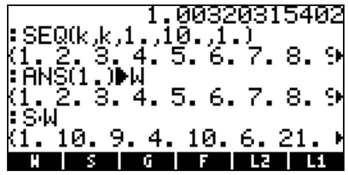


Wenn wir eine Liste mit Daten $\{s_1, s_2, \ldots, s_n\}$ und eine Liste mit Gewichten $\{w_1, w_2, \ldots, w_n\}$ haben, kann das gewogene Mittel der Daten in S wie folgt definiert werden:

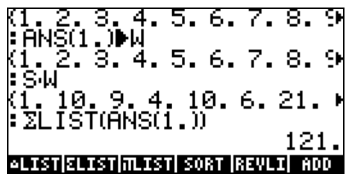
$$s_w = \frac{\sum_{k=1}^{n} w_k \cdot s_k}{\sum_{k=1}^{n} w_k}.$$

Um nun das gewogene Mittel der Daten aus der Liste S mit den Werten der Gewichte aus der Liste W zu berechnen, können wir wie folgt vorgehen:

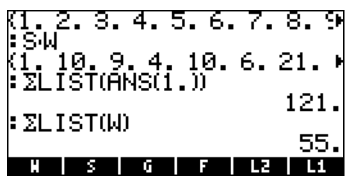
1. Multiplizieren Sie die Listen S und W:



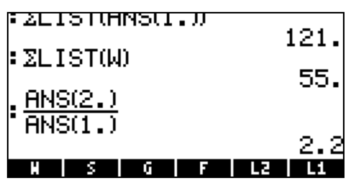
2. Wenden Sie die Funktion ΣLIST auf das erzielte Ergebnis an um den Zähler s_w zu berechnen:



3. Wenden Sie die Funktion ΣLIST ein weiteres Mal an, um den Nenner von s_w: zu berechnen:



4. Verwenden Sie den Ausdruck ANS(2)/ANS(1), um das gewogene Mittel zu berechnen:



Somit ist das gewogene Mittel einer Liste S mit Gewichten in Liste W gleich s_w= 2,2.

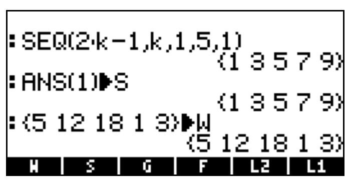
> **Anmerkung:** ANS(1) bezieht sich auf das neueste Ergebnis (55), während ANS(2) sich auf das vorhergehende Ergebnis (121) bezieht.

Statistiken gruppierter Daten

Gruppierte Daten werden normalerweise als Tabelle, unter Angabe der Frequenz (w) der Daten in Klassen oder Bins angezeigt. Jede Klasse oder Bin wird durch eine Klassenmarke (s), normalerweise der Mittelpunkt der Klasse, repräsentiert. Nachfolgend ein Beispiel von gruppierten Daten:

Klasse Grenzen	Klasse Marke s_k	Häufigkeit Zähler w_k
0 - 2	1	5
2 - 4	3	12
4 - 6	5	18
6 - 8	7	1
8 -10	9	3

Die Daten der Klassenmarken können in der Variablen S gespeichert werden, während der Häufigkeitszähler in der Variablen W, wie folgt gespeichert werden kann:

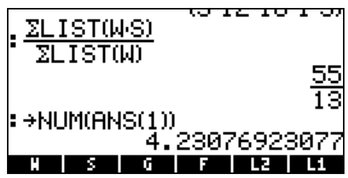


Angenommen unsere Liste von Klassenmarken ist S = {s_1, s_2, …, s_n } und die Liste der Häufigkeitszähler W = {w_1, w_2, …, w_n }, dann stellt der Mittelwert der Daten in S mit den Gewichten W den Mittelwert der gruppierten Daten dar, welche in diesem Kontext als s, bezeichnet werden.

$$\bar{s} = \frac{\sum_{k=1}^{n} w_k \cdot s_k}{\sum_{k=1}^{n} w_k} = \frac{\sum_{k=1}^{n} w_k \cdot s_k}{N},$$

wobei $N = \sum_{k=1}^{n} w_k$ den gesamten Häufigkeitszähler darstellt.

Der Mittelwert der Daten in Liste S und W, kann somit mit nachfolgender Prozedur, wie oben für den gewogenen Mittelwert hervorgehoben, berechnet werden, d.h.

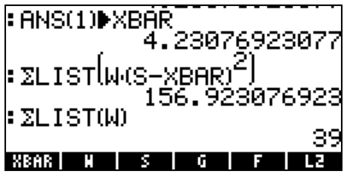


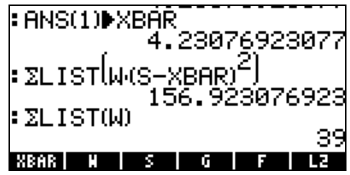
Wir speichern diesen Wert in einer Variablen mit den Namen XBAR:

Die Abweichung dieser gruppierten Daten wird wie folgt definiert

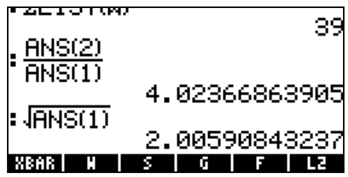
$$V = \frac{\sum_{k=1}^{n} w_k \cdot (s_k - \bar{s})^2}{\sum_{k=1}^{n} w_k} = \frac{\sum_{k=1}^{n} w_k \cdot (s_k - \bar{s})^2}{N}$$

Um das letzte Ergebnis zu berechnen, können wir wie folgt vorgehen:





Die Standardabweichung der gruppierten Daten ist die Quadratwurzel der Abweichung:



Kapitel 9
Vektoren

Dieses Kapitel stellt Beispiele zur Eingabe und Operation mit Vektoren zur Verfügung, für beide, den mathematischen bestehend aus vielen Elementen, aber auch den physischen bestehend aus nur 2 bis 3 Komponenten.

Definitionen

Aus mathematischer Sicht ist ein Vektor eine Gruppierung von 2 oder mehr in einer Spalte oder Zeile angeordnete Elemente. Diese bezeichnen wir als Zeilen- oder Spaltenvektoren. Nachfolgend werden einige Beispiele gezeigt:

$$v = \begin{bmatrix} -1 \\ 3 \\ 6 \end{bmatrix}, \quad u = [1, -3, 5, 2]$$

Physikalische Vektoren haben zwei oder drei Komponenten und können zur Darstellung von physikalischen Dimensionen wie: Position, Geschwindigkeit, Beschleunigung, Kraft, Momente, Impulse und Drehimpulse, Winkelgeschwindigkeit und Drehbeschleunigung usw. eingesetzt werden. Auf das Kartesische Koordinatensystem (x,y,z) bezogen, gibt es Einheitsvektoren **i**, **j**, **k** die den einzelnen Richtungen der Koordinaten zugeordnet sind, so dass ein physikalischer Vektor hinsichtlich seiner Komponenten wie folgt geschrieben werden kann A_x, A_y, A_z als **A** = A_x**i** + A_y**j** + A_z**k**.
Eine alternative Notation für diese Vektoren ist: **A** = [A_x, A_y, A_z], **A** = (A_x, A_y, A_z), oder **A** = < A_x, A_y, A_z >. Eine zweidimensionale Version dieses Vektors wird als **A** = A_x**i** + A_y**j**, **A** = [A_x, A_y], **A** = (A_x, A_y) oder **A** = < A_x, A_y > dargestellt. Da im Rechner die Vektoren zwischen eckigen Klammern [ ] geschrieben werden, wählen wir von nun an die Schreibweise **A** = [A_x, A_y, A_z] or **A** = [A_x, A_y, A_z], um auf zwei- oder dreidimensionale Vektoren hinzuweisen. Die Magnitude eines Vektors **A** wird als |**A**| = $\sqrt{A_x^2 + A_y^2 + A_z^2}$ definiert. Ein Einheitsvektor in Richtung des Vektors **A** wird als $\mathbf{e}_A$ = **A**/|**A**| definiert. Vektoren können mit einer Skalarzahl multipliziert werden, z.B. k**A** = [kA_x, kA_y, kA_z]. Physikalisch gesehen ist der Vektor k**A**

parallel zu Vektor **A** wenn k>0 oder antiparallel zu Vektor **A**, wenn k<0 ist. Die Negative eines Vektors wird als –**A** = (–1)**A** = [–A_x, –A_y, –A_z] definiert. Division durch eine Skalarzahl kann als Multiplikation interpretiert werden, d.h. **A**/k = (1/k)·**A**. Addition und Subtraktion von Vektoren wird als **A**±**B** = [$A_x \pm B_x$, $A_y \pm B_y$, $A_z \pm B_y$] definiert, wobei **B** den Vektor **B** = [B_x, B_y, B_z] darstellt.
Es gibt zwei Definitionen von Produkten von physikalischen Vektoren, ein Skalar oder internes Produkt (Skalarprodukt) und ein Vektor oder äußeres Produkt (Kreuzprodukt). Das Skalarprodukt erzeugt einen Skalarwert, welcher als **A•B** = |**A**||**B**|cos(θ) definiert wird, wobei θ den Winkel zwischen den beiden Vektoren darstellt. Das Kreuzprodukt erzeugt einen Vektor **A**×**B** dessen Magnitude |**A**×**B**| = |**A**||**B**|sin(θ) ist, und dessen Richtung durch die sogenannte Dreifingerregel bestimmt wir (für eine grafische Darstellung dieses Vorgangs, sehen Sie in einem Physik-, Mathe- oder Mechanik-Buch nach. Hinsichtliche Kartesischer Komponenten ist **A•B** = $A_xB_x+A_yB_y+A_zB_z$ und **A**×**B** = [A_yB_z-A_zB_y,A_zB_x-A_xB_z,A_xB_y-A_yB_x]. Der Winkel zwischen zwei Vektoren kann aus der Definition des Skalarproduktes als cos(θ) = **A•B**/|**A**||**B**|= $\mathbf{e}_A \bullet \mathbf{e}_B$ ermittelt werden. Somit, wenn zwei Vektoren **A** und **B** senkrecht zueinander stehen ist (θ = 90^0 = $\pi/2^{rad}$), **A•B** = 0.

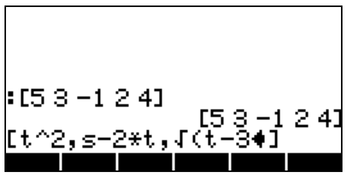
Eingabe von Vektoren

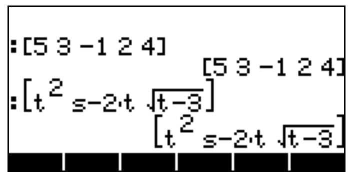
Im Rechner werden die Vektoren als eine, in Klammer eingeschlossene, Reihe von Zahlen dargestellt und typischerweise als Reihe von Vektoren eingegeben. Im Rechner werden die Klammern mit der Tastenkombination ← [] erstellt, welche der Taste × zugeordnet ist. Nachfolgend einige Beispiele von Vektoren im Rechner.

`[3.5, 2.2, -1.3, 5.6, 2.3]`	Eine allgemeine Reihe von Vektoren
`[1.5,-2.2]`	Ein 2-D Vektor
`[3,-1,2]`	Ein 3-D Vektor
`['t','t^2','SIN(t)']`	Ein Vektor von algebraischen Objekten

Eingabe von Vektoren in den Stack

Ist der Rechner im ALG-Modus, wird der Vektor durch öffnen eines Klammerpaares (← []) und eintippen der Komponenten oder Elemente innerhalb dieser, durch Komma getrennt (↱ ,), eingegeben. Die nachfolgenden Abbildungen zeigen die Eingabe eines numerischen, gefolgt von einem algebraischen Vektor. Die linke Abbildung zeigt den algebraischen Vektor vor drücken der Taste ↵. Die Abbildung rechts zeigt die Anzeige des Rechners, nachdem der algebraische Vektor eingegeben wurde:



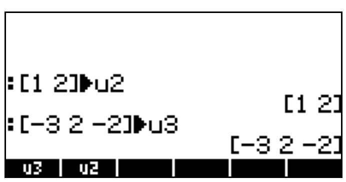


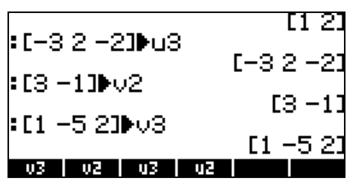
Im RPN-Modus können Sie einen Vektor in den Stack eingeben, indem Sie ein Klammernpaar öffnen und die Komponenten oder Elemente des Vektors entweder durch Komma (↱ ,) oder Leerzeichen (SPC) getrennt eingeben. Beachten Sie, dass nachdem Sie die Taste ENTER gedrückt haben, der Rechner in beiden Fällen die Elemente des Vektors durch Leerzeichen getrennt anzeigt.

Vektoren in Variablen speichern

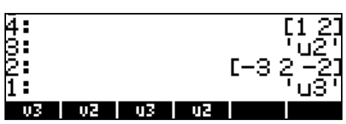
Vektoren können in Variablen gespeichert werden. In den folgenden Abbildungen sehen Sie die Vektoren

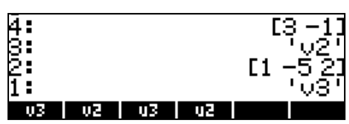
$\mathbf{u}_2$ = [1, 2], $\mathbf{u}_3$ = [-3, 2, -2], $\mathbf{v}_2$ = [3,-1], $\mathbf{v}_3$ = [1, -5, 2] entsprechend in den Variablen u2, u3, v2 und v3 gespeichert. Als erstes im ALG-Modus:





Dann im RPN-Modus (bevor Sie wiederholt die Taste STO▸ drücken):





Eingabe von Vektoren mit Hilfe des MatrixWriters (MTRW)

Vektoren können auch über den MatrixWriter ‹MTRW , eingegeben werden (dritte Taste vierte Reihe von oben). Dieser Befehl erzeugt eine Art Tabelle, welche den Reihen und Spalten einer Matrix entspricht (Details zur Anwendung und Benutzung des MatrixWriters werden in einem nachfolgenden Kapitel erörtert). Für einen Vektor möchten wir Daten nur in die oberste Reihe eingeben. Standardmäßig ist die Zelle der ersten Zeile und Spalte ausgewählt. Am unteren Teil der Tabelle finden Sie nachfolgende Funktionstasten:

EDIT VEC■ ←WID WID→ GO→■ GO↓

Benutzen Sie die Taste EDIT, um die Inhalte einer ausgewählten Zelle des MatrixWriters zu ändern.

Wenn ausgewählt, wird die Taste VEC einen Vektor, im Gegensatz zu einer Matrix, von einer Zeile und mehreren Spalten, erzeugen.

Vektoren vs. Matrizen

Um zu sehen wie die Taste VEC funktioniert, machen Sie folgende Übung:

(1) Starten Sie den MatrixWriter (‹MTRW). VEC■ und GO→■ ausgewählt, geben Sie ein wie folgt 3 ENTER 5 ENTER 2 ENTER ENTER. Dies erzeugt [3. 5. 2.]. Im RPN-Modus können Sie nachfolgende Tastenfolge benutzen, um das gleiche Ergebnis zu erhalten: 3 SPC 5 SPC 2 ENTER ENTER).

(2) VEC nicht ausgewählt und GO→■ ausgewählt, geben Sie ein wie folgt 3 SPC 5 SPC 2 ENTER ENTER. Dies erzeugt [[3. 5. 2.]].

Obwohl diese beiden Ergebnisse sich nur in der Anzahl der verwendeten Klammern unterscheiden, stellen diese für den Rechner unterschiedliche mathematische Objekte dar. Ersteres ist ein Vektor mit 3 Elementen und das zweite stellt eine Matrix mit einer Zeile und drei Spalten dar. Es gibt drei Unterschiede in der Art wie mathematische Operationen auf einen Vektor im Gegensatz zur Matrix angewendet werden. Deshalb behalten wir die Funktionstaste VEC■ selektiert, während wir den MatrixWriter benutzen.

Die Taste ←WID wird verwendet, um die Breite der Spalten in der Tabelle zu verringern. Drücken Sie die Taste mehrmals, um zu sehen wie sich die Spaltenbreite im MatrixWriter verringert.
Die Taste WID→ wird verwendet, um die Breite der Spalten in der Tabelle zu vergrößern. Drücken Sie die Taste mehrmals, um zu sehen wie sich die Spaltenbreite in ihrem MatrixWriter vergrößert.
Wenn ausgewählt, wählt die Taste GO→■ beim drücken der Taste ENTER automatisch die nächste Zelle rechts von der Position der aktuellen Zelle. Diese Option ist standardmäßig eingestellt.
Wenn ausgewählt, wählt die Taste GO↓ beim drücken der Taste ENTER automatisch die nächste Zelle unten von der Position der aktuellen Zelle.

Nach rechts bewegen vs. nach unten bewegen im MatrixWriter
Aktivieren Sie den MatrixWriter und geben Sie wie folgt ein
3 ENTER 5 ENTER 2 ENTER ENTER, die Taste GO→■ ist ausgewählt (Standard).
Um den Unterschied zu sehen, geben Sie als nächstes die gleiche Zahlenfolge, mit ausgewählter GO↓■ Taste ein. Im ersten Fall haben Sie einen Vektor bestehend aus drei Elementen eingegeben. Im zweiten Fall haben Sie eine Matrix bestehend aus drei Zeilen und einer Spalte eingegeben

Starten Sie den MatrixWriter über ← MTRW und drücken Sie NXT, um die zweite Zeile des Funktionsmenüs am unteren Teil der Anzeige, anzuzeigen. Es zeigt die Tasten:

+ROW -ROW +COL -COL ←STK GOTO

Die Taste +ROW wird eine Zeile mit lauter Nullen an der Stelle der ausgewählten Zelle in der Tabelle hinzufügen.
Die Taste -ROW wird die ganze Zeile, in der sie eine Zelle ausgewählt haben, löschen.
Die Taste +COL wird eine ganze Spalte Nullen an der Stelle der ausgewählten Zelle der Tabelle eintragen.
Die Taste -COL wird die Spalte in der sie eine Zelle ausgewählt haben, löschen.

Die Taste ↦STK wird den Inhalt der ausgewählten Zelle in den Stack verschieben.

Wenn die Taste GOTO gedrückt ist, wird der Benutzer aufgefordert die Zahl für die Zeile und Spalte, an die der Cursor positioniert werden soll, einzugeben.

Wird die Taste NXT ein weiteres Mal gedrückt, erscheint das letzte Menü, welches nur noch die eine Funktion DEL (löschen) enthält.

Die Funktion DEL wird die Inhalte der ausgewählten Zelle löschen und diese mit einer Null ersetzen.

Um zu sehen wie diese Tasten funktionieren, machen Sie folgende Übung:

(1) Starten Sie den MatrixWriter über ← MTRW . Stellen Sie sicher, dass die Tasten VEC■ und GO→■ ausgewählt sind.

(2) Geben Sie folgendes ein:

1 ENTER 2 ENTER 3 ENTER
NXT GOTO 2 OK 1 OK OK
2 ENTER 1 ENTER 5 ENTER
4 ENTER 5 ENTER 6 ENTER
7 ENTER 8 ENTER 9 ENTER

(3) Bewegen Sie den Cursor zwei Positionen nach oben indem Sie die Pfeiltaste ▲ zweimal drücken. Drücken Sie anschließend -ROW. Die zweite Zeile wird verschwinden.

(4) Drücken Sie nun +ROW. Eine Zeile von drei Nullen erscheint nun in der zweiten Zeile.

(5) Drücken Sie -COL. Die erste Spalte verschwindet.

(6) Drücken Sie +COL. Eine Zeile von zwei Nullen erscheint nun in der ersten Zeile.

(7) Drücken Sie nun GOTO 3 OK 3 OK OK, um zu Position (3,3) zu springen.

(8) Drücken Sie ↦STK. Der Inhalt der Zelle (3,3) wird in den Stack verschoben, obwohl Sie diese im Moment noch nicht sehen können.

(9) Drücken Sie ENTER Dies sollte eine Null an Position (3,3) eintragen, aber diese Funktion scheint trotzdem einwandfrei zu funktionieren.

Verwendung des MatrixWriters zur Eingabe von Vektoren

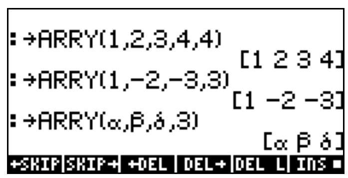
Zusammengefasst, um einen Vektor anhand des MatrixWriters einzugeben, starten Sie diesen (← MTRW) und geben Sie die Elemente des Vektors ein, indem Sie nach jedem einzelnen Element die Taste ENTER drücken. Drücken Sie anschließend ENTER ENTER . Stellen Sie sicher, dass die Tasten VEC■ und WID→■ ausgewählt sind.

Beispiel: ← MTRW ' ALPHA ← X Y^x 2 ENTER 2 ENTER 5 +/- ENTER ENTER

erzeugt: ['x^2' 2 –5]

Erstellen eines Vektors mit Hilfe von →ARRY

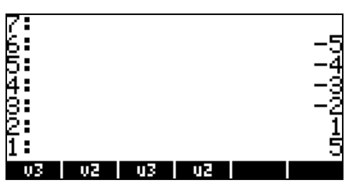
Auch die Funktion →ARRY, welche im Funktionskatalog (→ CAT → ⟶, verwenden Sie die Pfeiltasten ▲▼, um die Funktion zu finden), zu finden ist, kann zur Eingabe von Vektoren verwendet werden. Geben Sie im ALG-Modus →ARRY(*Elemente des Vektors, Anzahl der Elemente*) ein, z.B.

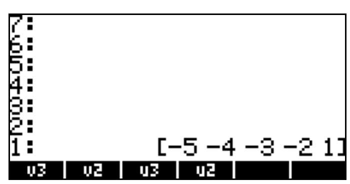


Im RPN-Modus:

(1) Geben Sie die n Elemente in der Reihenfolge, wie Sie diese angezeigt haben möchten, (wenn von links nach rechts gelesen) in den RPN Stack ein.
(2) Geben Sie *n* als letzten Eintrag ein.
(3) Verwenden Sie die Funktion →ARRY.

Ihre Anzeige wird im RPN Stack wie folgt angezeigt – vor und nach anwenden der Funktion →ARRY:



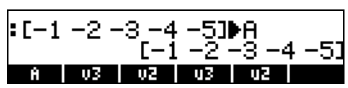


Im RPN-Modus, nimmt die Funktion [→ARRY] die Objekte aus den Stack Ebenen *n+1, n, n-1*, …, bis hin zu Ebenen 3 und 2 und konvertiert diese in einen Vektor bestehend aus *n* Elementen. Das Objekt das sich ursprünglich in Stack Ebene n+1 befindet, wird so zum ersten Element, das Objekt aus Ebene n das zweite Element und so weiter.

Anmerkung: Die Funktion →ARRY kann auch über das Menü PRG/TYPE (← PRG) aufgerufen werden.

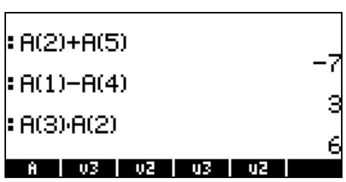
Kennung, Extraktion und Hinzufügen von Elementen des Vektors

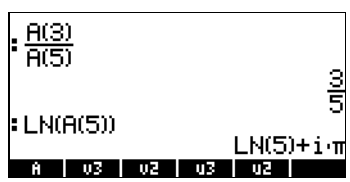
Speichern Sie einen Vektor in einer Variablen, sagen wir A, können Sie die Elemente des Vektor kennzeichnen, indem Sie A(i) verwenden, wobei i eine Integerzahl, kleiner oder gleich der Vektorgröße darstellt. Erstellen Sie z.B. nachfolgendes Array (Reihe) und speichern Sie dieses in der Variablen A: [-1, -2, -3, -4, -5]:

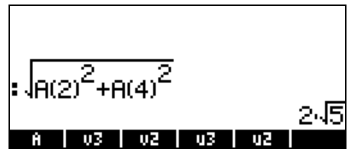


Zur Erinnerung das dritte Element in A, z.B. könnten Sie als A(3) in den Rechner eingeben. Im ALG-Modus geben Sie einfach A(3) ein. Im RPN-Modus müssen Sie 'A(3)' ENTER EVAL eingeben.

Mit den Elementen des Arrays können Sie Operationen durchführen, indem Sie algebraische Ausdrücke eingeben und berechnen, wie z.B.:





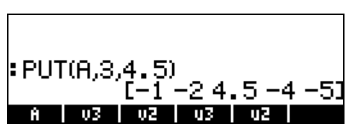


Sie können auch kompliziertere <u>*Ausdrücke, in denen die Elemente*</u> von A vorkommen, erstellen. So können wir z.B. mit Hilfe des EquationWriters (⮣ EQW) die folgende Summenbildung der Elemente aus A eingeben:

Markieren wir nun den gesamten Ausdruck und benutzen die Funktionstaste EVAL, erhalten wir das Ergebnis: -15.

Anmerkung: Vektor A können wir auch als *indexierte Variable* bezeichnen, weil A nicht nur einen, sondern mehrere Werte, welche durch den Unterindex identifiziert werden, darstellt.

Um ein <u>Element im Array zu ersetzen</u> verwenden wir die Funktion PUT (diese kann aus dem Funktionen Katalog ⮣ CAT oder dem Untermenü PRG/LIST/ELEMENTS - letzteres wurde in Kapitel 8 erläutert – aufgerufen werden). Im ALG-Modus müssen Sie die Funktion PUT mit folgenden Argumenten verwenden: PUT(*Array, zu ersetzende Position, neuer Wert*). So z.B., um den Inhalt von A(3) auf 4,5 zu ändern, geben Sie wie folgt ein:



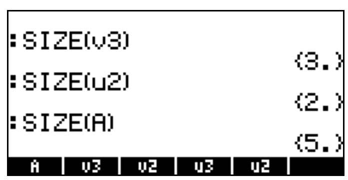
Im RPN-Modus können sie den *Wert eines Elementes* aus A *ändern*, indem Sie einen neuen Wert in diesem Element speichern. Wenn wir z.B. den Inhalt von A(3) von seinem derzeitigen Wert -3 auf 4,5 ändern möchten, gehen wir wie folgt vor:

4 . 5 ENTER ' ALPHA A ← () 3 ENTER STO▶

Um diese Änderung zu überprüfen drücken wir: ↱ . Das Ergebnis sieht nun wie folgt aus: [-1 -2 4.5 -4 -5].

Anmerkung: Dieser Ansatz den Wert eines Elementes im Array zu ändern, ist im ALG-Modus nicht erlaubt; versuchen Sie den Wert 4,5 in A(3) in diesem Modus zu speichern, erhalten Sie nachfolgende Fehlermeldung: Invalid Syntax (ungültige Syntax).

Die Länge eines Vektors können Sie über die Funktion SIZE ermitteln – diese kann über dem Befehle Katalog (N) oder über das Untermenü PRG/LIST/ELEMENTS aufgerufen werden. Nachfolgend einige Beispiele, die auf vorhin gespeicherte Arrays und Vektoren basieren:

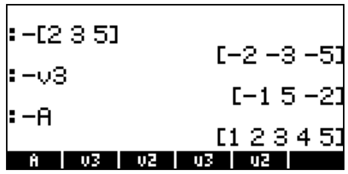


Einfache Operationen mit Vektoren

Um Operationen mit Vektoren zu veranschaulichen, verwenden wir die Vektoren A, u2, u3, v2 und v3, die wir in einer vorangegangenen Übung gespeichert haben.

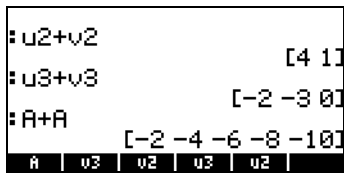
Änderung des Vorzeichens

Um das Vorzeichen eines Vektors zu ändern, benutzen Sie die Taste +/-, z.B.



Addition, Subtraktion

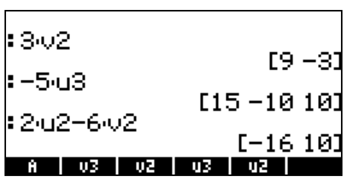
Bei der Addition und Subtraktion von Vektoren müssen die beiden Operanden des Vektors die gleiche Länge haben:

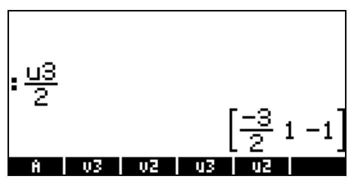


Ein Versuch Vektoren verschiedener Länge zu addieren oder zu subtrahieren, erzeugt eine Fehlermeldung (Invalid Dimension), so z.B. v2+v3, u2+u3, A+v3, usw.

Multiplikation und Division mit einem Skalar

Multiplikation und Division mit einem Skalar ist ganz einfach:

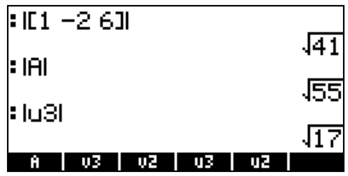




Funktion Absoluter Wert

Wird die Funktion Absoluter Wert (ABS) auf einen Vektor angewandt, ermittelt diese die Magnitude des Vektors. Die Magnitude für einen Vektor A = $[A_1, A_2, \ldots, A_n]$ wird wie folgt definiert $|A| = \sqrt{A_x^2 + A_y^2 + \cdots + A_z^2}$. Im ALG-Modus geben Sie den Namen der Funktion, gefolgt von den Argumenten des

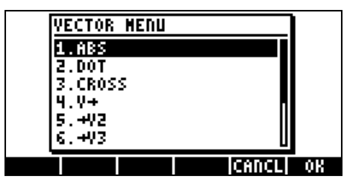
Vektors, ein. So zum Beispiel: wird der Ausdruck ABS([1,–2,6]), ABS(A), ABS(u3), in der Anzeige wie folgt aussehen:



Das Menü MTH/VECTOR

Das Menü MTH (⬅ MTH) enthält ein für Objekte von Vektoren spezielles Funktionsmenü.

Das Menü VECTOR enthält die folgenden Funktionen (System Flag 117 ist auf CHOOSE Kästchen gesetzt):

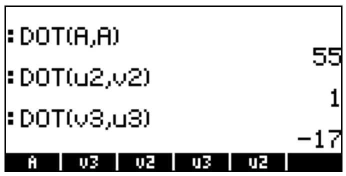


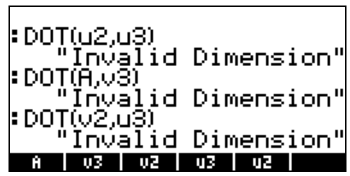
Magnitude (Größenordnung)

Die Magnitude eines Vektors, wie vorher beschrieben, kann mit der Funktion ABS gefunden werden. Die Funktion steht auch über die Tastatur (⬅ ABS) zur Verfügung. Anwendungsbeispiele für die Funktion ABS wurden vorher gezeigt.

Skalarprodukt

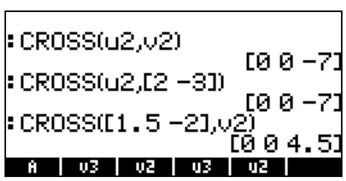
Die Funktion DOT wird zur Berechnung des Skalarproduktes zweier Vektoren der gleichen Länge verwendet. Einige Beispiele zur Anwendung der Funktion DOT, unter Verwendung der vorher gespeicherten Vektoren A, u2, u3, v2, and v3, werden als nächstes im ALG-Modus gezeigt. Der Versuch das Skalarprodukt zweier Vektoren unterschiedlicher Länge zu berechnen, führt zu einer Fehlermeldung:

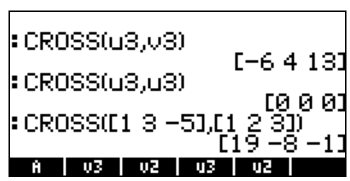




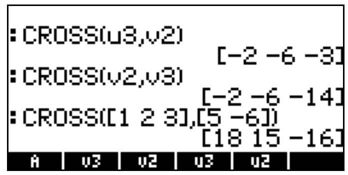
Kreuzprodukt

Die Funktion CROSS wird zur Berechnung des Kreuzproduktes zweier 2-D Vektoren, zweier 3-D Vektoren oder eines 2-D und eines 3-D Vektors, eingesetzt. Um ein Kreuzprodukt zu berechnen, wird ein 2-D Vektor der Form [A_x, A_y] als 3-D Vektor [A_x, A_y,0] behandelt. Nachfolgend werden Beispiele zweier 2-D und zweier 3-D Vekoren im ALG-Modus angezeigt. Beachten Sie, dass das Kreuzprodukt zweier 2-D Vektoren einen Vektor nur in z-Richtung, d.h. einen Vektor der Form [0, 0, C_z] erzeugt.





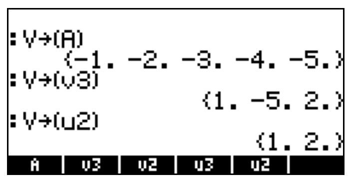
Beispiele eines Kreuzproduktes eines 3-D Vektors mit einem 2-D Vektor, oder umgekehrt, werden nachfolgend gezeigt:



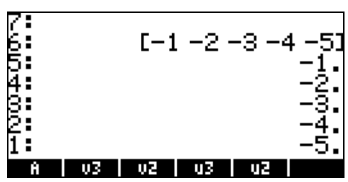
Der Versuch ein Kreuzprodukt zweier Vektoren deren Länge nicht 2 oder 3 ist, wird eine Fehlermeldung erzeugen: (Invalid Dimension), z.B. CROSS(v3,A), usw.

Zerlegung eines Vektors

Zur Zerlegung eines Vektors in seine Elemente und Komponenten wird die Funktion V→ verwendet. Wird diese im ALG-Modus benutzt, erzeugt V→ eine Liste mit den Elementen des Vektors, z.B.

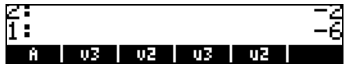


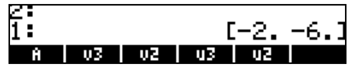
Wenn im RPN-Modus angewendet, listet die Funktion V→ die Liste der Elemente im Stack auf, z.B erstellt V→(A) die folgende Ausgabe im RPN Stack (Vektor A wird in Ebene 6 angezeigt).



Erstellen eines zweidimensionalen Vektors

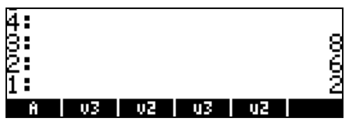
Die Funktion →V3 wird im RPN-Modus zur Erstellung eines Vektors mit den Werten in Stack Ebene 1: und 2: verwendet. Ihre Anzeige, vor und nach Anwenden der Funktion →V2, wird wie folgt aussehen:

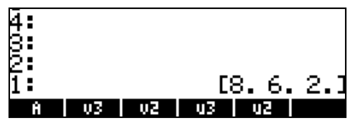




Erstellen eines dreidimensionalen Vektors

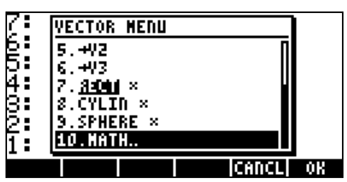
Die Funktion →V2 wird im RPN-Modus zur Erstellung eines Vektors mit den Werten in Stack Ebene 1:, 2: und 3: verwendet. Ihre Anzeige, vor und nach Anwenden der Funktion →V2, wird wie folgt aussehen:



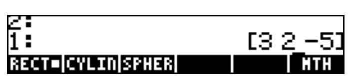


Änderung des Koordinatensystems

Um das aktuelle Koordinatensystem in ein rechtwinkliges (Kartesisches), zylindrisches (Polar) oder sphärisches zu ändern, werden die Funktionen RECT, CYLIN und SPHERE verwendet. Das aktuelle System wird entsprechend markiert in CHOOSE Kästchen (System Flag 117 nicht gesetzt) oder als ausgewählt im SOFT Menü (System Flag 117 gesetzt) angezeigt. In der nachfolgenden Abbildung sehen Sie die Darstellung des RECT (rechtwinkligen) Koordinatensystems in diesen beiden Formaten:



Ist das rechtwinklige oder Kartesische Koordinatensystem ausgewählt, erscheint in der oberen Zeile des Displays ein Feld XYZ, und alle 2-D oder 3-D Vektoren, die sich in Ihrem Rechner befinden, als Komponenten (x,y,z) des Vektors dargestellt. Um den Vektor A = 3**i**+2**j**-5**k** einzugeben, verwenden wir die Werte [3,2,-5]; der Vektor wird wie folgt angezeigt:



Um anstelle einer Kartesischen Komponente eines Vektors eine zylindrische (Polar) Komponente einzugeben, müssen wir die Magnitude r zur Verfügung stellen, die Projektion des Vektors auf die x-y Ebene, einen Winkel θ (im aktuellen Winkelmaß), welcher die Neigung von r auf die positive x-Achse darstellt, sowie eine Z-Komponente des Vektors. Dem Winkel θ muss das Winkelzeichen (∠) vorangesetzt sein, erzeugt mit den Tasten [ALPHA][↱][6]. Nehmen wir z.B. an, dass wir einen Vektor r = 5, θ = 25° (DEG sollte als Winkelmaß ausgewählt sein) und z = 2,3 haben, können wir den Vektor wie folgt eingeben:

[↰][[]] [5] [↱][,] [ALPHA][↱][6] [2][5] [↱][,] [2][.][3]

Bevor Sie [ENTER] drücken, sieht die Anzeige wie die links dargestellte Abbildung aus. Nach drücken der Taste [ENTER], sieht Ihre Anzeige wie in der rechten Abbildung aus (Das numerische System wurde auf Fix mit drei Dezimalstellen geändert).

Beachten Sie, dass der Vektor in Kartesischen Koordinaten angezeigt wird, mit den Komponenten x = r cos(θ), y = r sin(θ), z = z, obwohl wir diese in Polar-Koordinatensystem eingegeben haben.Das ist deshalb so, weil die Anzeige des Vektors immer im standardmäßig eingestellten Koordinatensystem erfolgt. In diesem Fall haben wir x = 4,532, y = 2,112 und z = 2,300

Angenommen wir geben nun einen Vektor ins sphärische Koordinatensystem ein (d.h. als (ρ,θ,ϕ), wobei ρ die Länge des Vektors darstellt, θ der Winkel der xy Projektion des Vektors auf die positiven Seite der x-Achse ist und ϕ der von ρ und der positiven Seite der z-Achse Winkel darstellt) mit den Werten ρ = 5, θ = 25° und ϕ = 45°. Wir verwenden dazu: [↰][[]] [5] [↱][,] [ALPHA][↱][6] [2][5] [,] [ALPHA][↱][6] [4][5]

Die nachfolgende Abbildung zeigt die Umwandlung des Vektors von sphärische in Kartesische Koordinaten, mit x = ρ sin(ϕ) cos(θ), y = ρ sin (ϕ) cos (θ), z = ρ cos(ϕ). In diesem Fall ist x = 3,204, y = 1,494 und z = 3,536

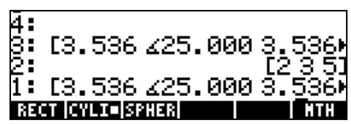
Wenn das CYLINdrical (zylindrische) System gewählt wurde, erscheint in der obersten Zeile des Displays ein Feld R∠Z und ein in zylindrischen Koordinaten eingegebener Vektor mit dessen zylindrischen (Polar) Koordinaten als (r,θ,z). Um dies zu veranschaulichen, ändern wir das Koordinatensystem auf CYLINdrical (zylindrisch), um zu sehen wie sich der Vektor in der letzten Anzeige in seine zylindrischen (Polar) Koordinaten ändert.Die zweite Komponente wird mit dem Winkelzeichen vor der Zahl (um deren rechtwinklige Eigenschaften zu betonen) dargestellt.

Die Konvertierung von Kartesischen auf zylindrische Koordinaten ist so, dass r = $(x^2+y^2)^{1/2}$, θ = $\tan^{-1}(y/x)$ und z = z, darstellt. Im obigen Fall sieht die Transformation folgendermaßen aus, (x,y,z) = (3,204, 2,112, 2,300) ergibt (r,θ,z) = (3,536,25°,3,536).

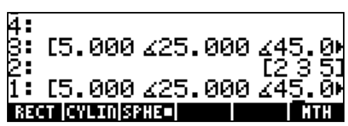
Ändern Sie an dieser Stelle das Winkelmaß auf Radian. An diesem Punkt ändern Sie das eckige Maß zu den Einheitswinkeln.
Geben wir nun einen Vektor in Kartesischer Form ein, obwohl das CYLINdrical (zylindrischen) Koordinatensystem aktiv ist, wir dieser in Kartesischen Koordinaten angezeigt, so z.B.

Das ist, weil die Integerzahlen alle für die Anwendung mit dem CAS gedacht sind, und somit die Komponenten des Vektors in Kartesischer Form erhalten bleiben.Um eine Konvertierung in Polar-Koordinaten zu erzwingen, geben Sie die Komponenten des Vektors als reelle Zahlen ein (d.h. Sie fügen einen Dezimalpunkt hinzu), so z.B. [2., 3., 5.].

Ist das zylindrische Koordinatensystem ausgewählt und wir einen Vektor mit sphärischen Koordinaten eingeben, wird dieser automatisch in seine zylindrischen (Polar) Äquivalente (r,θ,z) geändert, wobei r = ρ sin φ, θ = θ, z = ρ cos ist. Nachfolgend sehen Sie ein Beispiel eines Vektors, der mit sphärischen Koordinaten eingegeben und in seine Polar-Koordinaten umgewandelt wurde. In diesem Fall ρ = 5, θ = 25° und φ = 45°, während die Umwandlung anzeigt, dass r = 3,563 und z = 3,536 ist.



Als nächstes ändern wir das Koordinatensystem auf sphärische Koordinaten, indem wir die Funktion SPHERE aus dem Untermenü VECTOR des Menüs MTH verwenden. Sobald dieses Koordinatensystem ausgewählt wurde, wird in der obersten Zeile des Displays das R∠∠ Format angezeigt. Die letzte Anzeige wird sich wie folgt ändern:



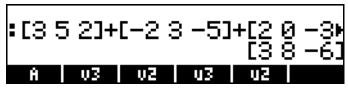
Beachten Sie, dass Vektoren die wir in zylindrischen (Polar) Koordinaten eingegeben haben, auf sphärisch umgeändert wurden. Die Umwandlung besteht darin, dass $\rho = (r^2+z^2)^{1/2}$, θ = θ und $\phi = \tan^{-1}(r/z)$ ist. Der Vektor der ursprünglich in Kartesischen Koordinaten dargestellt wurde, bleibt unverändert.

Anwendungen von Vektor-Operationen

In diesem Abschnitt zeigen wir Ihnen einige Beispiele von Vektor Operationen, die in der Physik oder Mechanik verwendet werden.

Resultante von Kräften

Angenommen ein Teilchen wird nachfolgenden Kräften (in N) ausgesetzt: $\mathbf{F}_1 = 3\mathbf{i}+5\mathbf{j}+2\mathbf{k}$, $\mathbf{F}_2 = -2\mathbf{i}+3\mathbf{j}-5\mathbf{k}$ und $F_3 = 2\mathbf{i}-3\mathbf{k}$.Um die Resultante zu ermitteln, d.h. die Summe all dieser Kräfte, können Sie im ALG-Modus folgenden Ansatz verwenden:



Somit ist die Resultante $R = \mathbf{F}_1 + \mathbf{F}_2 + \mathbf{F}_3 = (3\mathbf{i}+8\mathbf{j}-6\mathbf{k})$N. Im RPN-Modus verwenden Sie:

[3,5,2] ENTER [-2,3,-5] ENTER [2,0,3] ENTER + +

Winkel zwischen den Vektoren

Der Winkel zwischen zwei Vektoren **A, B** kann mit Hilfe der Formel $\theta = \cos^{-1}(\mathbf{A}\bullet\mathbf{B}/|\mathbf{A}||\mathbf{B}|)$ ermittelt werden.

Angenommen Sie wollen den Winkel zwischen den Vektoren $\mathbf{A} = 3\mathbf{i}-5\mathbf{j}+6\mathbf{k}$, $\mathbf{B} = 2\mathbf{i}+\mathbf{j}-3\mathbf{k}$ ermitteln, können Sie im ALG-Modus wie folgt vorgehen (Winkelmaß auf Grad eingestellt):

1 – Geben Sie die Vektoren [3,-5,6] ein, drücken Sie ENTER, [2,1,-3] dann ENTER.

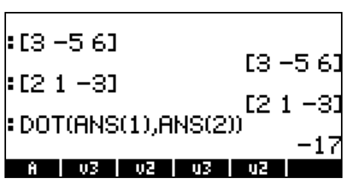
2 - DOT(ANS(1),ANS(2)) berechnet das Skalarprodukt

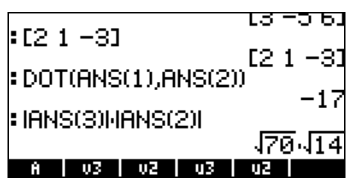
3 - ABS(ANS(3))*ABS((ANS(2)) berechnet Produkt der Magnituden

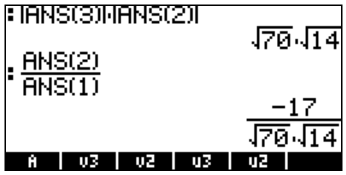
4 - ANS(2)/ANS(1) berechnet cos(θ)

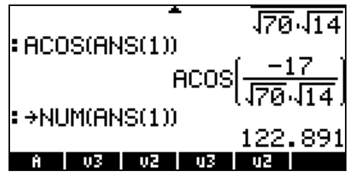
5 - ACOS(ANS(1)) gefolgt von →NUM(ANS(1)), berechnet θ

Die Schritte werden in den nachfolgenden Abbildungen (natürlich im ALG-Modus) angezeigt:







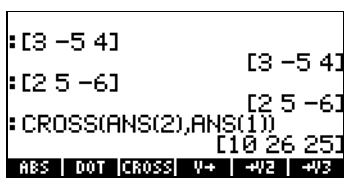


Dies ergibt das Ergebnis θ = 122,891°. Im RPN-Modus gehen Sie wie folgt vor:

[3,-5,6] ENTER [2,1,-3] ENTER DOT
[3,-5,6] ENTER ABS [2,1,-3] ENTER ABS ×
÷ ACOS →NUM

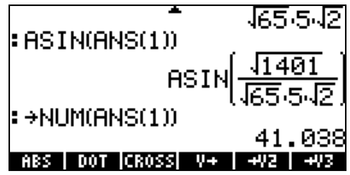
Kraftmoment

Das Moment das von einer Kraft **F** auf einen Punkt O ausgeübt wird, wird als Kreuzprodukt **M** = **r**×**F** bezeichnet, wobei **r**, auch als Kraftarm bekannt ist und die Position des Vektors in Punkt O in Richtung des Anwendungspunktes der Kraft darstellt. Angenommen eine Kraft **F** = (2**i**+5**j**-6**k**) hat einen Kraftarm von **r** = (3**i**-5**j**+4**k**)m. Um den Moment, den diese Kraft auf den Arm ausübt zu ermitteln, verwenden wir die Funktion CROSS, wie nachfolgend gezeigt:



Somit ist **M** = (10**i**+26**j**+25**k**) m·N. Wir wissen, dass die Mangitude von **M** sich so verhält, dass |**M**| = |**r**||**F**|sin(θ), wobei θ der Winkel zwischen **r** und **F** darstellt. Wir können diesen Winkel als θ = $\sin^{-1}$(|**M**| /|**r**||**F**|), über nachfolgende Operationen ermitteln:

1 – ABS(ANS(1))/(ABS(ANS(2))*ABS(ANS(3)) berechnet sin(θ)
2 – ASIN(ANS(1)), gefolgt von →NUM(ANS(1)) berechnet θ
Diese Rechenvorgänge werden in den nachfolgenden Abbildungen im ALG-Modus dargestellt:

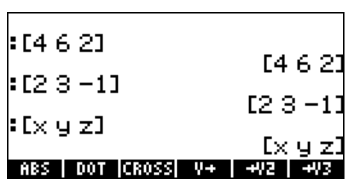


Somit beträgt der Winkel zwischen den Vektoren **r** und **F** θ = 41,038°. Im RPN-Modus können wir wie folgt vorgehen: `3,-5,4]` ENTER `[2,5,-6]` ENTER `CROSS` `ABS` `[3,-5,4]` ENTER `ABS` `[2,5,-6]` ENTER `ABS` × ÷ `ASIN` `→NUM`

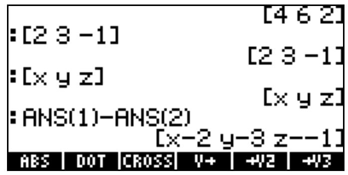
Gleichung einer Ebene im Raum

Nehmen wir an, dass wir einen Punkt $P_0(x_0,y_0,z_0)$ im Raum haben und einen Vektor **N** = N_x**i**+N_y**j**+N_z**k** normal zu einem Punkt auf dieser Ebene, welche den Punkt P_0 enthält hat, unser Problem ist es die Gleichung für diese Ebene zu finden. Wir können einen Vektor mit dem Startpunkt P_0 und dem Endpunkt an Position P(x,y,z), ein willkürlicher Punkt auf dieser Ebene, erstellen.Somit ist dieser Vektor **r** = P_0P = $(x\text{-}x_0)$**i**+ $(y\text{-}y_0)$**j** + $(z\text{-}z_0)$**k** senkrecht auf den normalen Vektor **N**, da **r** sich komplett in der Ebene befindet. Wir haben gesehen, dass für zwei normale Vektoren **N** und **r**, **N•r** =0 ist. Dadurch können wir dieses Ergebnis zur Ermittlung der Gleichung der Ebene verwenden.

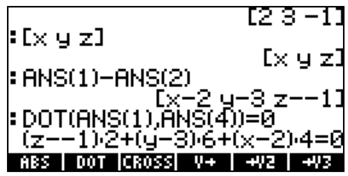
Um diesen Ansatz zu veranschaulichen, nehmen wir an der Punkt P_0 ist $P_0(2,3,\text{-}1)$ und der Normalvektor **N** = 4**i**+6**j**+2**k,** dann können wir den Vektor **N** und den Punkt P_0 als zwei Vektoren, wie unten gezeigt, eingeben: Als letztes geben wir noch den Vektor [x,y,z] ein:



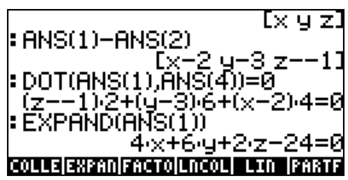
Dann berechnen wir den Vektor P_0P = **r** as ANS(1) – ANS(2), d.h.



Schließlich nehmen wir das Skalarprodukt von ANS(1) und ANS(4) und setzen dies gleich Null, um die Operation **N•r** =0 zu vervollständigen:



Nun können wir die Funktion EXPAND (im ALG-Menü) verwenden, um den Ausdruck zu expandieren (aufzufächern):



Somit ist die Gleichung der Ebene durch den Punkt $P_0(2,3,-1)$ mit einem normalen Vektor von **N** = 4**i**+6**j**+2**k**, 4x + 6y + 2z – 24 = 0. Im RPN-Modus verwenden Sie:

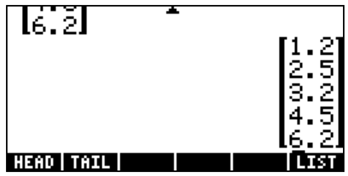
`[2,3,-1]` ENTER `['x','y','z']` ENTER – `[4,6,2] DOT EXPAND`

Zeilen- und Spaltenvektoren sowie Listen

Alle in diesem Kapitel gezeigten Vektoren sind Zeilenvektoren. In einigen Fällen, aber ist es notwendig Spaltenvektoren zu erstellen (z.B um vordefinierte statistische Funktionen im Rechner anzuwenden). Der einfachste Weg einen Spaltenvektor einzugeben, ist, jedes einzelne Element des Vektors in ein Klammerpaar zu setzen, alle zusammen befinden sich in einem weiteren Klammerpaar. So zum Beispiel, geben Sie ein:

`[[1.-{}-2],[2.5],[3.2],[4.5],[6.2]]` `ENTER`

Dies wird im nachfolgenden Spaltenvektor dargestellt:



In diesem Abschnitt zeigen wir Ihnen wie Sie einen Spalten- in einen Zeilenvektor, einen Zeilen- in einen Spaltenvektor, eine Liste in einen Vektor und einen Vektor (oder Matrix) in eine Liste umwandeln können.

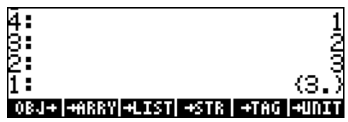
Als erstes zeigen wir diese Umwandlungen im RPN-Modus. In diesem Modus verwenden wir die Funktionen OBJ→, →LIST, →ARRY und DROP, um die Umwandlung durchzuführen. Um einen einfacheren Zugang zu diesen Funktionen zu bekommen, setzen wir das System Flag 117 auf SOFT Menüs (siehe Kapitel 1). Wenn dieses Flag gesetzt ist, können die Funktionen OBJ→, →ARRY und →LIST über `←` PRG `TYPE` aufgerufen werden. Die Funktionen OBJ→, →ARRY und →LIST können über die Funktionstasten `F1`, `F2` und `F3` aufgerufen werden. Auf die Funktion DROP kann über `←` PRG `STACK` `DROP` zugegriffen werden.

Nachfolgend erläutern wir die Anwendung der Funktionen OBJ→, →LIST, →ARRY und DROP mit einigen Beispielen.

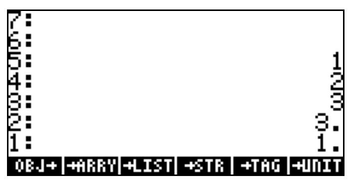
Funktion OBJ→

Diese Funktion zerlegt ein Objekt in seine Komponenten. Ist das Argument eine Liste, wird die Funktion OBJ→ die Elemente der Liste im Stack anzeigen, wobei die Anzahl der Elemente in Stack Ebene 1 angezeigt wird, so z.B. `{1,2,3}` `ENTER` `←` PRG `TYPE` `OBJ→` ergibt:

Wird die Funktion OBJ→ auf einen Vektor angewandt, wird eine Liste mit den Elementen des Vektors im Stack angezeigt, die Anzahl der Elemente des Vektors innerhalb von Klammern (eine Liste) in Ebene 1: Nachfolgendes Beispiel veranschaulicht diese Anwendung: [1,2,3] ENTER
← PRG TYPE OBJ→ ergibt:



Wenden wir nun die Funktion OBJ→ ein weiteres Mal an, wird die Liste {3.} in Stack Ebene 1: wie folgt zerlegt:



Funktion →LIST

Diese Funktion wird zur Erstellung einer Liste eingesetzt, wenn die Elemente der Liste und die Länge oder Größe der Liste bekannt ist. Im RPN Modus, sollte die Listengröße, sagen wir n, in Stack Ebene 1: eingegeben werden. Die Elemente der Liste sollten in die Stack Ebenen 2:, 3:, …, n+1: eingegeben werden. Um z.B. die Liste {1, 2, 3} zu erstellen, geben Sie folgendes ein:

1 ENTER 2 ENTER 3 ENTER 3 ENTER ← PRG TYPE →LIST.

Funktion →ARRY

Diese Funktion wird zur Erstellung eines Vektors oder einer Matrix verwendet. In diesem Abschnitt werden wir diese zu Erstellung eines Vektors oder eines Spaltenvektors (d.h. eine Matrix aus n Zeilen und einer Spalte) verwenden. Um einen regulären Vektor zu erstellen, tragen wir die Elemente des Vektors in den Stack und in Stack Ebene 1: geben wir die Vektorgröße als Liste, z.B.

1 ENTER 2 ENTER 3 ENTER ← {} 3 ENTER ← PRG TYPE →ARRY ein.

Um einen Spaltenvektor bestehend aus n Elementen zu erstellen, geben wir die Elemente des Vektors in den Stack und in Stack Ebene 1 die Liste {n 1} ein. So z.B. 1 ENTER 2 ENTER 3 ENTER ← {} 1 → , 3 ENTER ← PRG TYPE →ARRY.

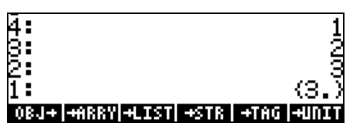
Funktion DROP

Diese Funktion hat die gleiche Wirkung wie die Löschtaste (◄).

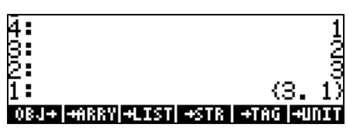
Umwandlung eines Zeilenvektors in einen Spaltenvektor

Wir veranschaulichden die Umwandlung mit dem Vektor [1,2,3]. Geben Sie diesen Vektor in den RPN Stack, um die Übung zu verfolgen.Um einen Zeilen- in einen Spaltenvektor umzuwandeln, müssen wir die nachfolgenden Operationen im RPN Stack durchführen:

1 - den Vektor mit der Funktion OBJ→ zerlegen



2 - +1 drücken, um die Liste in Stack Ebene 1: von {3} auf {3,1} umzuwandeln

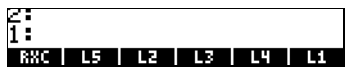


3 - die Funktion →ARRY verwenden, um den Spaltenvektor zu erzeugen

Wir können diese drei Schritte in ein UserRPL Programm, wie nachfolgend (im RPN-Modus, immer noch) gezeigt, eingeben:

→ « » ← PRG TYPE OBJ→ 1 + →ARRY
ENTER ' ALPHA ALPHA R X C ENTER STO▶

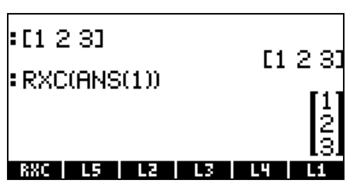
Eine neue Variable, RXC, wird nach drücken von VAR im Funktionsmenü zur Verfügung stehen:



Drücken Sie → RXC, um das in der Variablen RCX enthaltene Programm anzuzeigen:

<< OBJ→ 1 + →ARRY >>

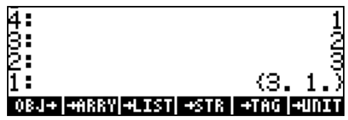
Diese Variable, RXC, kann nun zur direkten Umwandlung eines Zeilenvektors in einen Spaltenvektor verwendet werden.Geben Sie im RPN-Modus den Zeilenvektor ein und drücken Sie anschließend RXC. Versuchen Sie z.B.: [1,2,3] ENTER RXC.
Nachdem wir nun diese Variable definiert haben, können wir diese auch im ALG-Modus dazu verwenden einen Zeilen- in einen Spaltenvektor umzuwandeln.Ändern wir nun den Modus des Rechners auf ALG und versuchen wir nachfolgende Prozedur: [1,2,3] ENTER VAR RXC ← () ← ANS , ergibt dann:



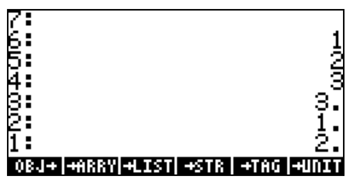
Umwandlung eines Spaltenvektors in einen Zeilenvektor

Um diese Umwandlung zu veranschaulichen, geben wir den Spaltenvektor `[[1],[2],[3]]` im RPN-Modus ein.Gehen Sie dann, wie in nachfolgender Übung gezeigt, vor, um den Spalten- in einen Zeilenvektor umzuwandeln.

1 - verwenden Sie die Funktion OBJ→, um den Spaltenvektor zu zerlegen

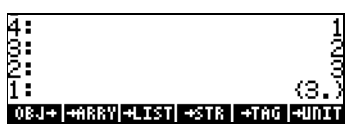


2 - verwenden Sie die Funktion OBJ→, um die Liste in Stack Ebene 1: zu zerlegen



3 – drücken Sie die Löschatste ◄ (auch als Funktion DROP bekannt), um die Zahl aus Stack Ebene 1: zu eliminieren

4 – werwenden Sie die Funktion →LIST, um eine Liste zu erstellen

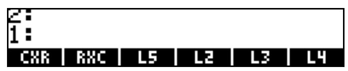


5 – verwenden Sie die Funktion →ARRY , um den Zeilenvektor zu erzeugen

Wir können diese fünf Schritte in ein UserRPL Programm, wie nachfolgend (im RPN-Modus, immer noch) gezeigt, eingeben:

→ « » ← PRG TYPE OBJ→ OBJ→
← PRG STACK DROP ← PRG TYPE →LIST →ARRY ENTER
' ALPHA ALPHA C X R ENTER STO▸

Eine neue Variable, CXR, wird nach drücken von VAR im Funktionsmenü zur Verfügung stehen:



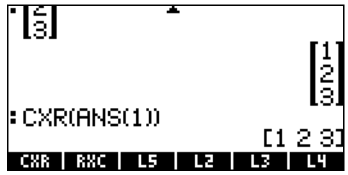
Drücken Sie → CXR, um das in der Variablen CXR enthaltene Programm anzuzeigen:

<< OBJ→ OBJ→ DROP →ARRY >>

Diese Variable, CXR, kann nun zur direkten Umwandlung eines Spaltenvektors in einen Zeilenvektor verwendet werden. Geben Sie im RPN-Modus den Zeilenvektor ein und drücken Sie anschließend CXR. Versuchen Sie z.B.: [[1],[2],[3]] ENTER CXR.
Nachdem wir nun die Variable CXR definiert haben, können wir diese auch im ALG-Modus dazu verwenden einen Zeilen- in einen Spaltenvektor umzuwandeln. Ändern wir nun den Modus des Rechners auf ALG und versuchen wir nachfolgende Prozedur:

[[1],[2],[3]] ENTER VAR CXR ← () ← ANS

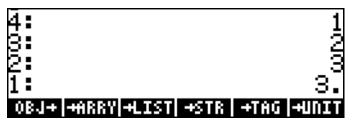
Das Resultat sieht dann so aus:



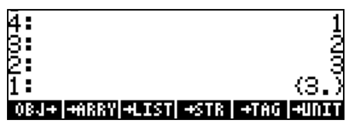
Eine Liste in einen Vektor umwandeln

Um diese Umwandlung zu veranschaulichen, geben wir die Liste {1,2,3} im RPN-Modus ein.Gehen Sie dann, wie in nachfolgender Übung gezeigt, vor, um die Liste in einen Vektor umzuwandeln.

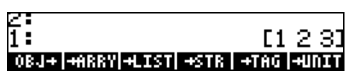
1 - verwenden Sie die Funktion OBJ→, um den Spaltenvektor zu zerlegen



2 - geben Sie eine 1 ein und verwenden dann die Funktion →LIST, um eine Liste in Stack Ebene 1: zu erstellen.



3 - verwenden Sie die Funktion →ARRY, um den Vektor zu erzeugen



Wir können diese drei Schritte in ein UserRPL Programm, wie nachfolgend (im RPN-Modus) gezeigt, eingeben:

→ « » ← PRG TYPE OBJ→ OBJ→ / →LIST →ARRY ENTER
' ALPHA ALPHA L X V ENTER STO▶

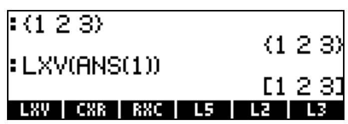
Eine neue Variable, LXV, wird nach drücken der Taste VAR unter den Funktionstasten zur Verfügung stehen.

Drücken Sie ↱ LXV, um das in der Variablen LVX enthaltene Programm anzuzeigen:

<< OBJ→ 1 →LIST →ARRY >>

Diese Variable, LXV, kann nun zur direkten Umwandlung einer Liste in einen Vektor verwendet werden.Geben Sie im RPN-Modus den Zeilenvektor ein und drücken Sie anschließend LXV. Versuchen Sie z.B.: {1,2,3} ENTER LXV.

Nachdem wir nun die Variable LXV definiert haben, können wir diese auch im ALG-Modus dazu verwenden eine Liste in einen Vektor umzuwandeln. Ändern wir nun den Modus des Rechners auf ALG und versuchen wir nachfolgende Prozedur: {1,2,3} ENTER VAR LXV ↰ () ↰ ANS , ergibt:



Einen Vektor (oder eine Matrix) in eine Liste umwandeln

Im Rechner wird die Funktion AXL zur Umwandlung eines Vektors in eine Liste bereitgestellt.Diese Funktion können Sie aus dem Befehle Katalog wie folgt aufrufen:

↱ CAT ALPHA ALPHA A X L ALPHA OK

Als Beispiel, wenden Sie die Funktion AXL auf den Vektor [1,2,3] im RPN-Modus, unter Verwendung von : [1,2,3] ENTER AXL an. Die nachfolgende Anzeige zeigt die Anwendung der Funktion AXL auf den gleichen Vektor im ALG-Modus.


:AXL([1 2 3])
{1 2 3}

Kapitel 10
Erstellen und manipulieren von Matrizen

In diesem Kapitel finden Sie Beispiele zur Erstellung von Matrizen im Rechner und zur Veranschaulichung der Manipulation von Matrizen-Elementen.

Definitionen

Eine Matrix ist ganz einfach ein rechtwinkliges Array von Objekten (d.h. Zahlen, Algebraiks) bestehend aus mehreren Zeilen und Spalten. Eine Matrix **A** mit n Zeilen und m Spalten, enthält somit n×m Elemente.Ein generisches Element einer Matrix wird durch die indexierte Variable a_{ij}, welche der Zeile i und Spalte j entspricht, dargestellt.Anhand dieser Notation können wir die Matrix **A** als **A** = $[a_{ij}]_{n\times m}$ definieren.Nachfolgend die gesamte Matrix:

$$\mathbf{A} = [a_{ij}]_{n\times m} = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1m} \\ a_{21} & a_{22} & \cdots & a_{2m} \\ \vdots & \vdots & \ddots & \\ a_{n1} & a_{n2} & \cdots & a_{nm} \end{bmatrix}.$$

Eine Matrix ist hermitisch (quadratisch) wenn m = n. Das Transponieren einer Matrix besteht darin die Zeilen mit den Spalten zu vertauschen und umgekehrt. Somit ist die Transponierte der Matrix **A**, $\mathbf{A}^T = [(a^T)_{ij}]_{m\times n} = [a_{ji}]_{m\times n}$. Die Hauptdiagonale einer hermitischen Matrix ist die Menge der Elemente a_{ii}. Eine Identitätsmatrix, $\mathbf{I}_{n\times n}$, ist eine hermitische Matrix deren Elemente der Hauptdiagonale alle 1, während alle anderen Elemente außerhalb der Diagonalen Null sind. So z.B. wird eine Identitätsmatrix 3×3 wie folgt definiert

$$\mathbf{I} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Auch kann eine Identitätsmatrix als $\mathbf{I}_{n\times n} = [\delta_{ij}]$ geschrieben werden, wobei δ_{ij}, eine Funktion, bekannt als Kronecker's Delta, darstellt und wie folgt definiert ist

$$\delta_{ij} = \begin{cases} 1, & if\ i = j \\ 0, & if\ i \neq j \end{cases}.$$

Eingaben von Matrizen in den Stack

In diesem Abschnitt zeigen wir zwei unterschiedliche Methoden, wie Matrizen in des Stack des Rechners eingegeben werden können: (1) mit Hilfe des MatrixEditors und (2) direkte Eingabe der Matrix in den Stack.

Verwendung des MatrixEditors

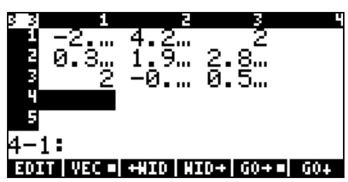
Genau wie Vektoren, in Kapitel 9 beschrieben, können Matrizen mit Hilfe des MatrixEditors in den Stack eingegeben werden.Um z.B. die Matrix einzugeben:

$$\begin{bmatrix} -2.5 & 4.2 & 2.0 \\ 0.3 & 1.9 & 2.8 \\ 2 & -0.1 & 0.5 \end{bmatrix},$$

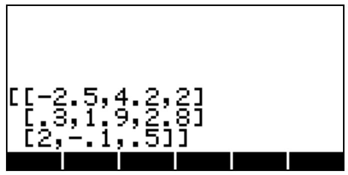
Starten Sie zuerst den MatrixEditor über ← MTRW .Stellen Sie sicher, dass die Option ⊞→■ ausgewählt ist.Verwenden Sie dazu die nachstehende Tastenfolge:

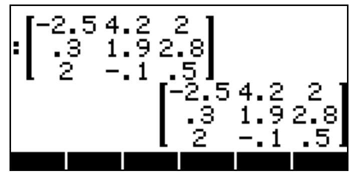
2 . 5 +/- ENTER 4 . 2 ENTER 2 ENTER ▼ ◄ ◄ ◄
. 3 ENTER 1 . 9 ENTER 2 . 8 ENTER
2 ENTER . 1 +/- ENTER . 5 ENTER

An dieser Stelle sieht Ihre Anzeige wie folgt aus:

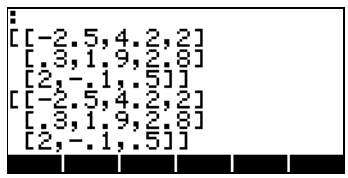


Drücken Sie ENTER ein weiteres Mal, um die Matrix in den Stack zu verschieben. Nachfolgend wird der Stack im ALG-Modus, vor und nach dem zweiten Mal drücken gezeigt:





Bei ausgewählter Textbuch Anzeige (über MODE DISP und ✓Textbook angekreuzt), wird die Matrix wie oben angezeigt: Andernfalls sieht die Matrix so aus:



Im RPN-Modus sieht die Anzeige in etwa gleich aus.

Anmerkung: Der MatrixWriter wurde in Kapitel 9 ausführlich erklärt.

Die Matrix direkt in den Stack eingeben

Das gleiche Ergebnis wie oben wird erzielt, wenn nachfolgendes direkt in den Stack eingeben wird:

← []
← [] 2 . 5 +/- → , 4 . 2 → , 2 ▶
→ ,
← [] . 3 → , 1 . 9 → , 2 . 8 ▶
→ ,
← [] 2 → , . 1 +/- → , . 5

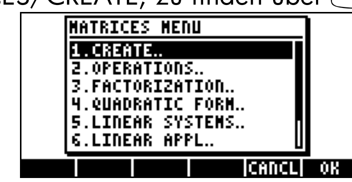
Um eine Matrix direkt in den Stack einzugeben, öffnen Sie ein Klammerpaar (← []) und setzen Sie jede Zeile der Matrix in ein weiteres Klammernpaar (← []) .Die Elemente der Matrix müssen durch Komma (→ , .) voneinander getrennt werden, genauso die Klammern zwischen den Zeilen. (**Anmerkung**: Im RPN-Modus können Sie die inneren Klammern nach Eingabe des ersten Zahlenpaares, weglassen, so können Sie z.B. anstelle von [[1 2 3] [4 5 6] [7 8 9]] einfach nur [[1 2 3] 4 5 6 7 8 9] eingeben.)

Speichern wir nun diese Matrix für spätere Übungen unter dem Namen A. Benutzen Sie dazu im ALG-Modus STO▸ ALPHA A .Und im RPN-Modus ' ALPHA A STO▸ .

Erstellen von Matrizen mit den Funktionen des Rechners

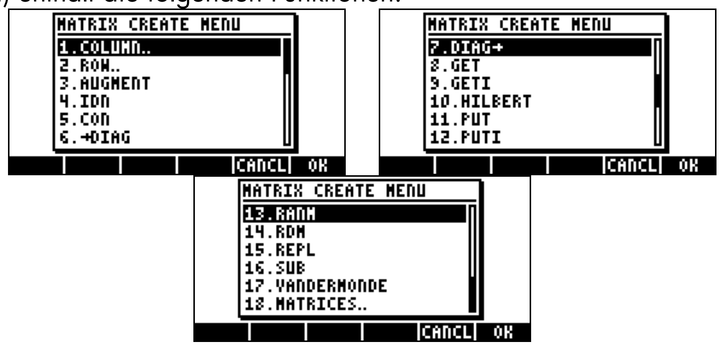
Manche Matrizen können mit den vorhanden Funktionen des Rechners, welche entweder im Untermenü MTH/MATRIX/MAKE oder innerhalb des MTH Menüs (↰ MTH) zur Verfügung stehen, erstellt werden,

oder im Menü MATRICES/CREATE, zu finden über ↰ MATRICES :



Das Untermenü MTH/MATRIX/MAKE (nennen wir es einfach das Menü MAKE) enthält die folgenden Funktionen:

Das Untermenü MATRICES/CREATE (nennen wir es einfach das Menü CREATE) enthält die folgenden Funktionen:

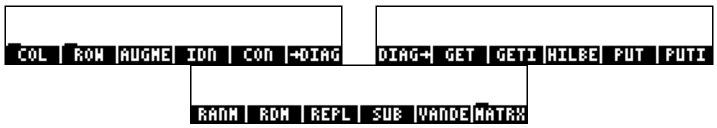


Wenn Sie die Menüs (MAKE und CREATE) näher betrachten, werden Sie feststellen, dass beide die gleichen Funktion enthalten GET, GETI, PUT, PUTI, SUB, REPL, RDM, RANM, HILBERT, VANDERMONDE, IDN, CON, →DIAG und DIAG→.Im Menü CREATE finden Sie die Untermenüs COLUMN (Spalte) und ROW (Zeile), welche Sie aber auch im Menü MTH/MATRIX finden. Im Menü MAKE sind die Funktionen SIZE enthalten, welche im Menü CREATE hingegen nicht enthalten sind.Im allgemeinen aber, stellen beide Menüs, MAKE und CREATE, die gleichen Funktionen zur Verfügung.In den nachfolgenden Beispielen erörtern wir, den Zugriff der Funktionen über das Menü MAKE.Am Ende dieses Abschnittes finden Sie eine Tabelle mit den entsprechenden Tastenfolgen, die dazu notwendig sind die gleichen Funktionen mit dem Menü CREATE zu erhalten, wenn das System Flag 117 auf SOFT Menü steht.

Ist das System Flag (Flag 117) auf SOFT Menü eingestellt, kann das Menü MAKE über die nachfolgende Tastenfolge gestartet werden: ← MTH MATRX MAKE

Die vorhandenen Funktionen werden als Funktionstasten des Menüs angezeigt (drücken Sie NXT, um den nächsten Satz von Funktionen anzuzeigen):

Wenn System Flag 117 auf SOFT Menüs steht, können die Funktionen des Menüs CREATE, über ← MATRICES CREAT ausgewählt werden und sehen wie folgt aus:

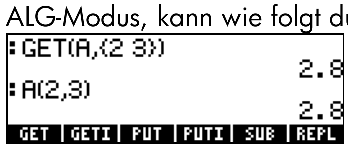


In den nächsten Abschnitten stellen wir die Anwendungen der Matrix-Funktionen im Menü MAKE und CREATE vor.

Funktionen GET und PUT

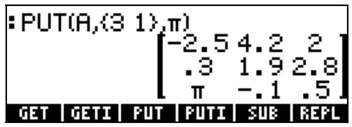
Die Funktionen GET, GETI, PUT und PUTI funktionieren mit Matrizen ähnlich wie mit Listen oder Vektoren, d.h. Sie müssen die Position der Elmente, welche Sie mit GET oder PUT verwenden wollen, angeben.Während aber in Listen und Vektoren nur ein Index zur Identifizierung eines Elementes notwendig ist, benötigen Sie in Matrizen zwei Indizes {Zeile, Spalte} zur Identifizierung der Elemente in der Matrix.Nachfolgend Anwendungsbeispiele für GET und PUT.

Nehmen wir die oben gespeicherte Matrix in die Variable A, um die Funktionen GET und PUT zu veranschaulichen.Die Extraktion des Elementes a_{23} aus der Matrix A im ALG-Modus, kann wie folgt durchgeführt werden:



Beachten Sie dabei, dass wir das gleiche Ergebnis erzielen, wenn wir A(2,3) eingeben und anschließend ENTER drücken. Im RPN-Modus geschieht das, wenn wir Sie A ENTER 3 ENTER GET oder A(2,3) ENTER eingeben.

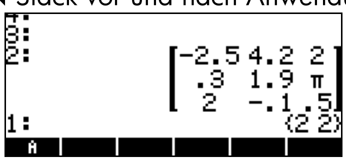
Angenommen wir möchten den Wert 'π' in Element a_{31} der Matrix eingeben. Dazu verwenden wir die Funktion PUT, so z.B.

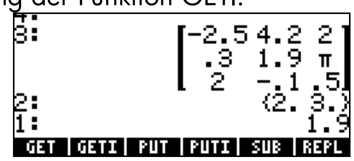


Im RPN-Modus können wir das folgendermaßen tun: VAR A {3,1} ENTER π PUT. Alternativ können wir im RPN-Modus auch nachfolgendes eingeben: π ' A(2,3) ENTER STO▸.Um den Inhalt der Variablen A anzuzeigen, drücken Sie A

Funktionen GETI und PUTI

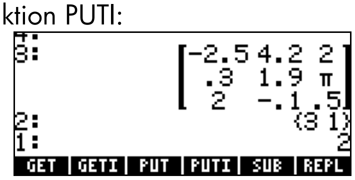
Die Funktionen PUTI und GETI werden in UserRPL Programmen verwendet, da sich diese einen Index für wiederholende Anwendungen von PUT und GET merken.Die Liste von Indizes in Matrizen variiert als Erstes in Spalten. Um die Anwendung zu veranschaulichen, schlagen wir nachfolgende Übung im RPN-Modus vor: A {2,2}ENTER GETI.Nachfolgende Abbildungen zeigen den RPN Stack vor und nach Anwendung der Funktion GETI:

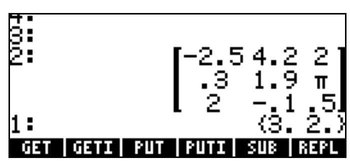




Beachten Sie dabei, dass die Abbildung für eine anschließende Anwendung der Funktion GETI oder GET zur Erhöhung des Spaltenindex der ursprünglichen Referenz um 1 (d.h. von {2,2} auf {2,3}) bereit ist, während gleichzeitig der extrahierte Wert, und zwar A(2,2) = 1,9, in Stack Ebene 1 angezeigt wird.

Nehmen wir nun an, dass Sie den Wert 2 in Element {3 1} mit Hilfe der Funktion PUTI eingeben möchten.Immer noch im RPN-Modus versuchen Sie es mit nachfolgenden Tastefolge: ◄ ◄{3 1} ENTER 2 ENTER PUTI. Nachfolgende Abbildungen zeigen den RPN Stack vor und nach Anwendung der Funktion PUTI:

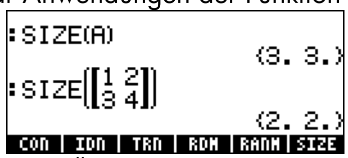




In diesem Fall wurde die 2 in Position {3 1} ersetzt, d.h. jetziger Wert A(3,1) = 2 und die Indexliste um 1 (Spalte zuerst), d.h. von {3,1} auf {3,2}.Die Matrix befindet sich in Ebene 2 und die um einen Schritt erhöhte Indexliste in Ebene 1.

Funktion SIZE

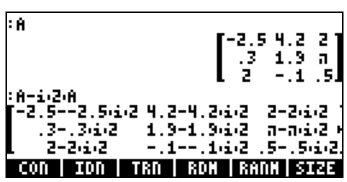
Die Funktion SIZE stellt eine Liste bereit, in welcher die Anzahl der Zeilen und Spalten der Matrix in Stack Ebene 1 angezeigt werden.Die nachfolgende Abbildung zeigt ein paar Anwendungen der Funktion SIZE im ALG-Modus:

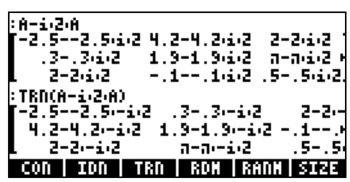


Im RPN-Modus, werden diese Übungen mit Hilfe von SIZE und [[1,2],[3,4]] ENTER SIZE durchgeführt.

Funktion TRN

Die Funktion TRN wird zur Erstellung der Transkonjugierten einer Matrix, d.h. der Tranponierten (TRAN) gefolgt von der Komplex-Konjugierten (CONJ) verwendet.In den nachfolgenden Abbildungen sehen Sie die ursprüngliche Matrix der Variablen A und deren Transponierte, zur vollständigen Ansicht in Kleinschrift angezeigt (siehe Kapitel 1).



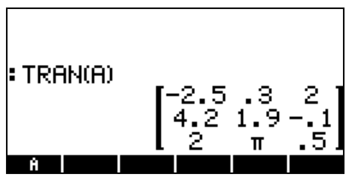


Ist das Argument eine reelle Matrix, erstellt TRN einfach die Transponierte der reellen Matrix.Versuchen Sie z.B. TRN(A) und vergleichen Sie dies mit TRAN(A).

Im RPN-Modus wird die Transkonjugierte einer Matrix A über TRN ermittelt.

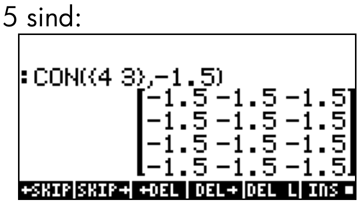
Anmerkung: Im Rechner steht auch die Funktion TRAN im Untermenü MATRICES/OPERATIONS zur Verfügung:

So z.B. im ALG-Modus:



Funktion CON

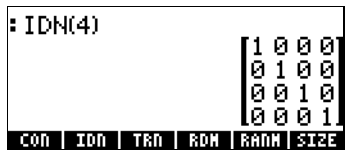
Das Argument der Funktion ist eine Liste von zwei Elementen, entsprechend der Anzahl der Zeilen und Spalten der zu erzeugenden Matrix und einem konstanten Wert.Die Funktion CON erstellt eine Matrix mit konstanten Elementen.So z.B. erzeugt der Befehl im ALG-Modus eine 4×3 Matrix, deren Elemente alle gleich -1,5 sind:



Im RPN-Modus wird das gleiche Ergebnis über {4,3} ENTER 1 . 5 +/- ENTER CON erreicht.

Funktion IDN

Die Funktion IDN (IDeNtity matrix) erstellt eine Identitätsmatrix von vorgegebener Größe.Beachten Sie, eine Identitätsmatrix muss eine hermitische Matrix sein, nur ein Wert wird benötigt, um diese komplett zu beschreiben. Um z.B. eine Identitätsmatrix von 4×4 im ALG-Modus zu erstellen, verwenden Sie:



Sie können aber genauso eine existierende hermitische Matrix als Argument der Funktion IDN verwenden, z.B.

Die erhaltene Identitätsmatrix wird die gleichen Dimensionen wie das Argument der Matrix haben.Beachten Sie dabei, dass der Versuch eine rechteckige Matrix (d.h. nicht hermitisch – quadratisch) als Argument von IDN zu erstellen, eine Fehlermeldung erzeugt.

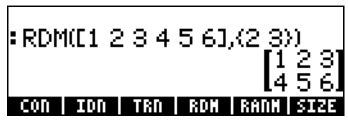
Im RPN-Modus, wurden die beiden obengenannten Beispiele mit Hilfe von 4 ENTER IDN and A IDN erzeugt.

Funktion RDM

Die Funktion RDM (Re-DiMensioning) wird zur Umschreibung von Vektoren und Matrizen als Matrizen und Vektoren verwendet.Die Eingabe zur Funktion besteht aus dem ursprünglichen Vektor oder der Matrize, gefolgt von einer Liste bestehend aus einer einzigen Zahl, wenn in einen Vektor konvertiert werden soll und aus zwei Zahlen, wenn die Konvertierung in eine Matrix erfolgen soll.Im vorangegangenen Fall stellt die Zahl die Dimension des Vektors dar, im letzteren Fall die Zahl der Zeilen und Spalten der Matrix. Die nachfolgenden Beispiele veranschaulichen die Anwendung der Funktion RDM:

Umdimensionieren eines Vektors in eine Matrix

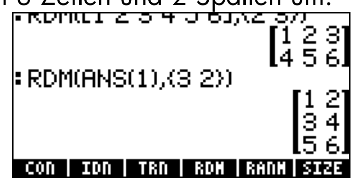
Die nachfolgenden Beispiele veranschaulichen wie ein Vektor bestehend aus 6 Elementen in eine Matrix von 2 Zeilen und 3 Spalten im ALG-Modus umdimensioniert wird:



Um die obige Matrix im RPN-Modus zu erstellen können wir dazu [1,2,3,4,5,6] `ENTER` {2,3} `ENTER` RDM verwenden.

Umdimensionieren einer Matrix in eine andere Matrix

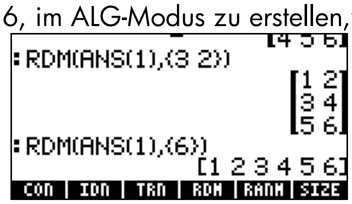
Im ALG-Modus, verwenden wir die oben erstellte Matrix und dimensionieren diese in eine Matrix von 3 Zeilen und 2 Spalten um:



Im RPN-Modus verwenden wir ganz einfach {3,2} `ENTER` RDM.

Umdimensionieren einer Matrix in einen Vektor

Um eine Matrix in einen Vektor umzudimensionieren, verwenden wir als Argumente die Matrix, gefolgt von einer Liste, die die Anzahl der Elemente in der Matrix enthält.So z.B., um die Matrix aus vorangegangenem Beispiel in einen Vektor der Länge 6, im ALG-Modus zu erstellen,verwenden Sie:

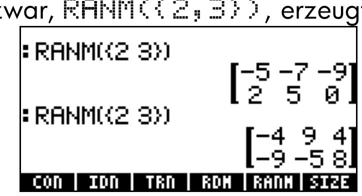


Im RPN-Modus, vorausgesetzt die Matrix befindet sich im Stack, verwenden wir {6} `ENTER` RDM.

Anmerkung: Die Funktion RDM stellt einen direkteren und effizienteren Weg zur Umwandlung von Listen in Arrays und umgekehrt dar, als die Möglichkeit, die am Ende von Kapitel 9 beschrieben wurde.

Funktion RANM

Die Funktion RANM (RANdom Matrix) erstellt eine Matrix mit zufälligen Integer Elementen, mit vorgegebener Liste der Anzahl der Zeilen und Spalten (d.h. die Dimensionen der Matrix).So werden z.B. im ALG-Modus zwei verschiede 2×3 Matrizen mit zufälligen Elementen, unter Verwendung des gleichen Befehls, und zwar, RANM({2,3}), erzeugt.

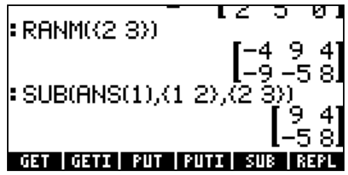


Im RPN-Modus verwenden Sie {2,3} ENTER RANM.

Offensichtlich werden die Ergebnisse aus Ihrem Rechner unterschiedlich von den oben gezeigten sein.Die generierten Zufallszahlen sind Integerwerte, gleichmäßig zwischen[-10,10] erzeugt, d.h. jede einzelne dieser 21 Zahlen kann mit derselben Wahrscheinlichkeit ausgewählt werden. Die Funktion RANM ist bei der Erstellung von Matrizen jeder Größe nützlich, um Operationen mit Matrizen oder der Anwendung von Matrix-Funktionen zu veranschaulichen.

Funktion SUB

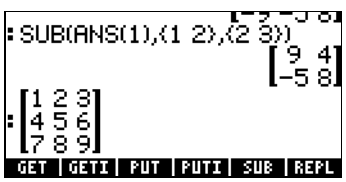
Die Funktion SUB extrahiert eine Untermatrix aus einer bestehenden Matrix, vorausgesetzt Sie geben den Anfangs- und Endwert der Untermatrix an.So z.B., wenn wir die Elemente a_{12}, a_{13}, a_{22} und a_{23} aus dem letzten Ergebnis, als eine Matrix 2×2 extrahieren wollen, verwenden wir im ALG-Modus:

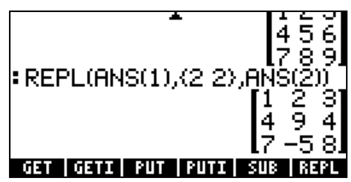


Im RPN-Modus, vorausgesetzt die ursprüngliche Matrix 2×3 befindet sich bereits im Stack, verwenden wir `{1,2}` ENTER `{2,3}` ENTER `SUB`.

Funktion REPL

Die Funktion REPL ersetzt oder fügt eine Untermatrix in eine größere Matrix ein. Die Eingabe für diese Funktion ist die Matrix, in welcher der Austausch erfolgen soll, die Position an welcher dieser Austausch zu erfolgen hat und die einzufügende Matrix. Als Beispiel - wir nehmen die gleiche Matrix, welche wir aus vorangegangenem Beispiel erhalten haben - geben wir die Matrix `[[1,2,3],[4,5,6],[7,8,9]]` ein.In der nachfolgenden linken Abbildung ist die neue Matrix im ALG-Modus vor drücken der Taste ENTER zu sehen. In der rechten Abbildung ist die Anwendung der Funktion RPL, zum ersetzen der Matrix 2×2 in `ANS(2)` in die Matrix 3×3, die sich im Moment in `ANS(1)` befindet, Anfangsposition in `{2,2}` zu sehen:



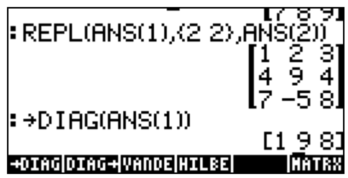


Wenn Sie im RPN-Modus arbeiten, vorausgesetzt die Matrix 2×2 befindet sich im Stack, gehen wir wie folgt vor:

`[[1,2,3],[4,5,6],[7,8,9]]` ENTER ▶ (diese letzte Taste vertauscht die Inhalte der Stack Ebene 1 mit 2) `{1,2}` ENTER ▶ (ein weiterer Austausch der Ebene 1 und 2) `REPL`.

Funktion →DIAG

Die Funktion →DIAG nimmt die Hauptdiagonale einer hermitischen Matrix mit den Dimensionen n×n und erstellt einen Vektor mit der Dimension n, der die Elemente der Hauptdiagonalen enhält.So z.B. für die Matrix, die uns aus vorangegangenem Beispiel bleibt, können wir die Hautpdiagonale wie folgt extrahieren:



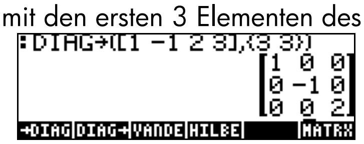
Im RPN-Modus, die Matrix 3×3 befindet sich im Stack, müssen wir einfach nur die Funktion →DIAG starten, um das gleiche Ergebnis wie oben zu erzielen.

Funktion DIAG→

Die Funktion DIAG→ nimmt einen Vektor und eine Liste von Matrix-Dimensionen {Zeilen, Spalten} und erstellt eine Diagonal-Matrix mit einer Hauptdiagonale, deren Elemente mit den richtigen Vektorelementen ersetzt wurde. So z.B. erzeugt der Befehl

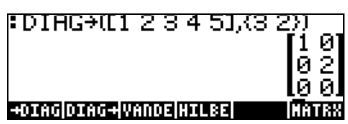
DIAG→([1,-1,2,3],{3,3})

eine Diagonale Matrix mit den ersten 3 Elementen des Vektor Argumentes:



Im RPN-Modus, können wir [1,-1,2,3] ENTER {3,3} ENTER DIAG→ verwenden, um das gleiche Ergebnis wie oben zu erzielen.

Ein weiteres Beispiel zur Anwendung der Funktion DIAG→ wird nun im ALG-Modus gezeigt:



Im RPN-Modus verwenden wir dafür [1,2,3,4,5] ENTER {3,2} ENTER DIAG→ .

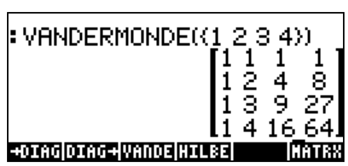
In diesem Fall soll eine 3×2 Matrix, mit so vielen Elementen des Vektors [1,2,3,4,5] wie möglich, als Hauptdiagonal-Elemente, erzeugt werden.Die Hauptdiagonale für eine rechtwinklige Matrix, startet in Position (1,1) und bewegt sich weiter zu (2,2), (3,3) usw. bis entweder die Anzahl der Zeilen oder der Spalten aufgebraucht ist. In diesem Fall, wurde die Anzahl der Spalten (2) vor der Anzahl der Zeilen (3) aufgebraucht, so dass die Hauptdiagonale nur die Elemente in den Positionen (1,1) und (2,2) enthält. Somit, wurden nur die ersten beiden Elemente des Vektors zur Bildung der Hauptdiagonale benötigt.

Funktion VANDERMONDE

Die Funktion VANDERMONDE erzeugt die Vandermonde-Matrix der Dimension n, basierend auf eine vorgegebene Liste von Eingabedaten.Die Dimension n stellt natürlich die Länge der Liste dar. Besteht die Eingabeliste aus den Objekten $\{x_1, x_2, \ldots x_n\}$, dann ist eine Vandermonde-Matrix im Rechner, eine Matrix, welche aus folgenden Elementen besteht:

$$\begin{bmatrix} 1 & x_1 & x_1^2 & \cdots & x_1^{n-1} \\ 1 & x_2 & x_2^2 & \cdots & x_2^{n-1} \\ 1 & x_3 & x_3^2 & \cdots & x_3^{n-1} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & x_n & x_n^2 & \cdots & x_n^{n-1} \end{bmatrix}$$

Als Beispiel geben Sie folgenden Befehl im ALG-Modus für die Liste {1,2,3,4} ein:



Im RPN-Modus geben Sie `{1,2,3,4}` ENTER `VANDERMONDE` ein.

Funktion HILBERT

Die Funktion HILBERT erstellt die Hilbert-Matrix für eine Dimension n.Die n×n Hilbert-Matrix $\mathbf{H}_n = [h_{jk}]_{n\times n}$, verhält sich wie folgt

$$h_{jk} = \frac{1}{j+k-1}$$

Die Hilbert-Matrix wird zur Anpassung numerischer Kurven durch die lineare Quadrat-Methode verwendet.

Programm zur Erstellung einer Matrix aus einer Anzahl von Listen

In diesem Abschnitt stellen wir ein paar UserRPL Programme, zur Erstellung einer Matrix aus einer Anzahl von Listen von Objekten, zur Verfügung. Die Listen können Spalten (Programm CRMC) oder Zeilen der Matrix (Programm CRMR) darstellen. Die Programme werden im RPN-Modus eingegeben und die Anweisungen der Tastenanschläge für System Flag 117 auf SOFT Menüs gesetzt. Dieser Abschnitt ist für Sie als Übung, in die Programmierfunktionen des Rechners zu gehen, gedacht. Die Programme werden unten aufgelistet, auf der linken Seite sind die zum Starten der Programmschritte notwendigen Tastenanschläge zu finden, während auf der rechten Seite, die Zeichen, die ins Display eingegeben werden, um jene Tastenanschläge durchzuführen. Als Erstes stellen wir die, zur Erstellung des Programms CRMC, notwendigen Schritte vor.

Die Listen stellen Spalten der Matrix dar

Das Programm CRMC ermöglicht Ihnen eine p×n Matrix (d.h. p Zeilen und n Spalten), jede bestehend aus n Einträgen von p Elementen, zu erstellen. Um dieses Programm zu erstellen verwenden Sie folgende Tastenaschläge:

Tastenfolge:	erzeugt:
↱ « »	«
↰ PRG STACK DUP	DUP
↱ → SPC ALPHA ↰ N	→ n

⟨→⟩ « »	<<
⟨1⟩ ⟨←⟩ PRG STACK SWAP	1 SWAP
⟨←⟩ PRG BRCH FOR FOR	FOR
⟨ALPHA⟩ ⟨←⟩ ⟨J⟩	j
⟨←⟩ PRG TYPE OBJ→	OBJ→
→ARRY	→ARRY
⟨←⟩ PRG BRCH IF IF	IF
⟨ALPHA⟩ ⟨←⟩ ⟨J⟩ ⟨SPC⟩	j
⟨ALPHA⟩ ⟨←⟩ ⟨N⟩	n
⟨←⟩ PRG TEST <	<
⟨←⟩ PRG BRCH IF THEN	THEN
⟨ALPHA⟩ ⟨←⟩ ⟨J⟩ ⟨SPC⟩ ⟨1⟩ ⟨+⟩	j 1 +
⟨←⟩ PRG STACK ⟨NXT⟩ ROLL	ROLL
⟨←⟩ PRG BRCH IF END	END
⟨←⟩ PRG BRCH FOR NEXT	NEXT
⟨←⟩ PRG BRCH IF IF	IF
⟨ALPHA⟩ ⟨←⟩ ⟨N⟩ ⟨SPC⟩ ⟨1⟩	n 1
⟨←⟩ PRG TEST >	>
⟨←⟩ PRG BRCH IF THEN	THEN
⟨1⟩ ⟨SPC⟩	1
⟨ALPHA⟩ ⟨←⟩ ⟨N⟩ ⟨SPC⟩ ⟨1⟩ ⟨−⟩	n 1 -
⟨←⟩ PRG BRCH FOR FOR	FOR
⟨ALPHA⟩ ⟨←⟩ ⟨J⟩ ⟨SPC⟩	j
⟨ALPHA⟩ ⟨←⟩ ⟨J⟩ ⟨SPC⟩ ⟨1⟩ ⟨+⟩	j 1 +
⟨←⟩ PRG STACK ⟨NXT⟩ ROLL	ROLL
⟨←⟩ PRG BRCH FOR NEXT	NEXT
⟨←⟩ PRG BRCH IF END	END
⟨ALPHA⟩ ⟨←⟩ ⟨N⟩ ⟨SPC⟩	n
⟨←⟩ MTH MATRX COL COL→	COL→
⟨ENTER⟩	Das Programm wird in Ebene 1 angezeigt

Zum sichern des Programms: ⟨'⟩ ⟨ALPHA⟩ ⟨ALPHA⟩ ⟨C⟩ ⟨R⟩ ⟨M⟩ ⟨C⟩ ⟨ALPHA⟩ ⟨STO▸⟩

Anmerkung: wenn Sie dieses Programm im HOME Verzeichnis speichern, kann dieses aus jedem Unterverzeichnis aus aufgerufen werden.

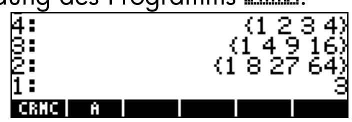
Um sich den Inhalt des Programms anzusehen drücken Sie ⟨VAR⟩ ⟨→⟩ CRMC. Das Programm-Listing sieht wie folgt aus:

```
« DUP → n « 1 SWAP FOR j OBJ→ →ARRY IF j n < THEN j 1 +
 ROLL END NEXT IF n 1 > THEN 1 n 1 - FOR j j 1 + ROLL
 NEXT END n COL→ » »
```

Um dieses Programm im RPN-Modus zu verwenden, geben Sie die n Einträge, in der Reihenfolge in welcher diese als Spalten der Matrix dargestellt werden sollen, ein, tragen Sie dann den Wert n ein und drücken Sie CRMC. Als Beispiel versuchen Sie nachfolgende Übung:

{1,2,3,4} ENTER {1,4,9,16} ENTER {1,8,27,64} ENTER 3 ENTER CRMC

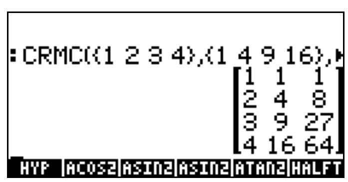
Ihre Anzeige wird im RPN Stack wie folgt angezeigt – vor und nach Anwendung des Programms CRMC:



Um dieses Programm im ALG-Modus zu verwenden, drücken Sie CRMC gefolgt von einem Klammerpaar (←()). Innerhalb der Klammern geben Sie die Einträge, welche die Spalten der Matrix darstellen, durch Kommas getrennt ein, und schließlich ein weiteres Komma und die Anzahl der Spalten. Der Befehl sollte wie folgt aussehen:

```
CRMC({1,2,3,4}, {1,4,9,16}, {1,8,27,64}, 3)
```

Nachfolgend ist Sie die Ausführung des Programms CRMC im ALG-Modus zu sehen:



Die Einträge stellen Spalten der Matrix dar

Das vorangegangene Programm kann leicht verändert werden, um eine Matrix zu erstellen, wenn die Eingabedaten Zeilen der fertigen Matrix werden sollen.Die einzige Änderung,die Sie vornehmen müssen, ist COL→ mit ROW→ im Programm-Listing auszutauschen.Um diese Änderung vorzunehmen, gehen Sie wie folgt vor:

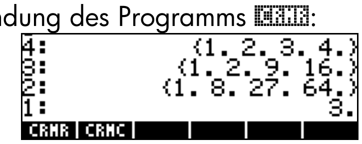
[→] CRMC	Das Programm CRMTC im Stack listen
[▼] [→] [▼] [▲] [◄] [◄] [◄]	ans Ende des Programms gehen
[⇐] [⇐] [⇐]	löschen von COL
[ALPHA] [ALPHA] [R] [O] [W] [ALPHA] [ENTER]	ROW eintippen, ins Programm gehen

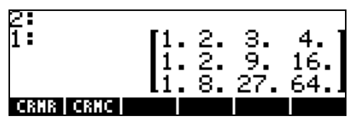
zur Speicherung des Programms verwenden Sie:

['] [ALPHA] [ALPHA] [C] [R] [M] [R] [ALPHA] [STO▶]

{1,2,3,4} [ENTER] {1,4,9,16} [ENTER] {1,8,27,64} [ENTER] 3 [ENTER] CRMR

Ihre Anzeige wird im RPN Stack wie folgt aussehen – vor und nach Anwendung des Programms CRMR:

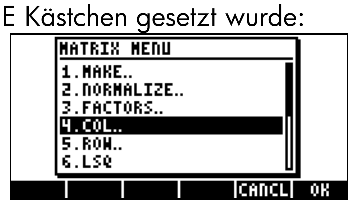




Diese Programme werden hauptsächlich bei statistischen Anwendungen verwendet, im Speziellen aber bei der Erstellung der Statistik-Matrix ΣDAT. Beispiele zur Anwendung dieses Programms, werden in einem späteren Kapitel gezeigt.

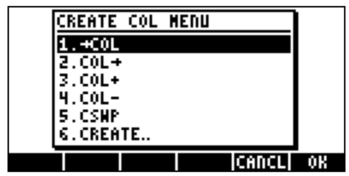
Manipulation der Spalten von Matrizen

Der Rechner enthält ein Menü mit Funktionen zur Manipulation von Matrizen in deren Spalten.Dieses Menü kann über MTH/MATRIX/COL.. ([←] MTH), wie in nachfolgender Abbildung, in Reihenfolge, gezeigt aufgerufen werden, während System Flag 117 auf CHOOSE Kästchen gesetzt wurde:

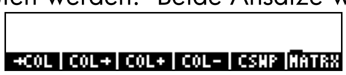


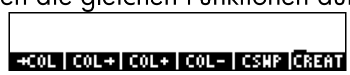
oder über das Untermenü MATRICES/CREATE/COLUMN:

Beide Ansätze weisen die gleichen Funktionen auf:



Ist das System Flag 117 auf SOFT Menüs gesetzt, kann das Menü COL entweder über ← MTH MATRX MAKE COL oder über ← MATRICES CREAT COL aufgerufen werden. Beide Ansätze weisen die gleichen Funktionen auf:

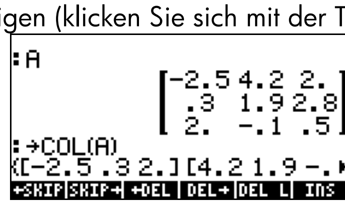


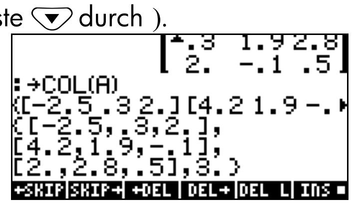


Die Anwendung dieser Funktionen wird nachfolgend dargestellt.

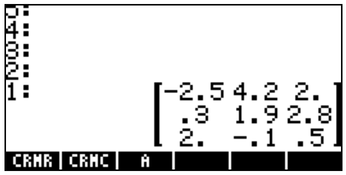
Funktion →COL

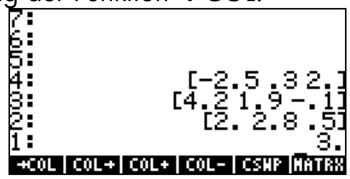
Die Funktion →COL nimmt als Argument eine Matrix und zerlegt diese in Vektoren, entsprechend ihrer Spalten.Nachfolgend wird eine Anwendung der Funktion →COL im ALG-Modus gezeigt.Die verwendete Matrix wurde zu einen früheren Zeitpunkt bereits in der Variablen A gespeichert. Die Matrix ist in der linken Abbildung zu sehen: Die rechte Abbildung zeigt die in Spalten zerlegte Matrix.Verwenden Sie den Zeileneditor, um das gesamte Ergebnis anzuzeigen (klicken Sie sich mit der Taste ▽ durch).





Im RPN-Modus müssen Sie die Matrix zuerst in den Stack laden, dann erst die Funktion →COL, d.h., →COL starten.Nachfolgende Abbildungen zeigen den RPN Stack vor und nach Anwendung der Funktion →COL.



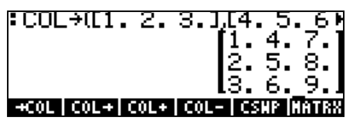


In diesem Ergebnis ist die erste Spalte, nach der Zerlegung, in der höchsten Stack Ebene, während in Stack Ebene 1 die Anzahl der Spalten der ursprünglichen Matrix zu finden ist.Die Matrix bleibt bei der Zerlegung nicht erhalten, d.h. sie ist im Stack nicht mehr verfügbar.

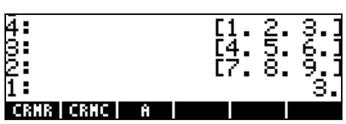
Funktion COL→

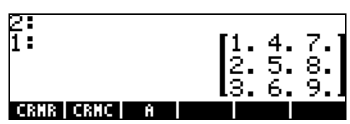
Die Funktion COL→ hat die entgegensetzte Wirkung der Funktion →COL, d.h. wenn Sie n Vektoren der gleiche Länge haben und die Zahl n, bildet die Funktion COL→ eine Matrix, indem sie die eingegebenen Vektoren als Spalten der Matrix darstellt. Hier ein Beispiel im ALG-Modus. Der verwendete Befehl war:

```
COL→([1,2,3],[4,5,6],[7,8,9],3)
```



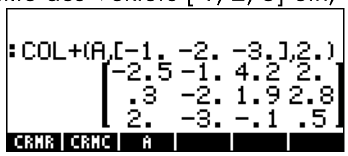
Geben Sie im RPN-Modus die n Vektoren in die Stack Ebenen n+1, n, n-1,…,2 ein und anschließend die Zahl n in Stack Ebene 1.So eingegeben, wird die Funktion COL→ die Vektoren als Spalten in die Matrix eingeben. Die nachfolgende Abbildung zeigt den RPN-Stack vor und nach Aktivierung der Funktion COL→.



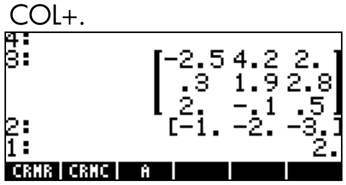


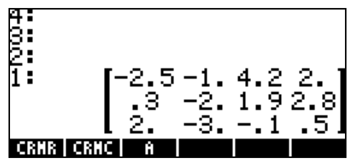
Funktion COL+

Die Funktion COL+ nimmt als Argument eine Matrix, einen Vektor der gleichen Länge wie die Anzahl der Zeilen in der Matrix und eine Integerzahl n, die die Position einer Spalte darstellt.Die Funktion COL+ fügt den Vektor in Spalte n der Matrix ein.Als Beispiel setzen wir im ALG-Modus die zweite Spalte in Matrix A mit Hilfe des Vektors [-1,-2,-3] ein, d.h.



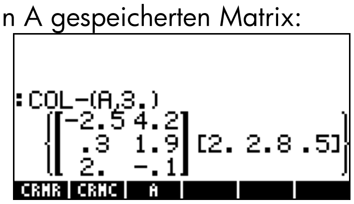
Im RPN-Modus geben wir zuerst die Matrix ein, dann den Vektor und die Nummer der Spalte, bevor wir die Funktion COL+ anwenden.Die nachfolgende Abbildung zeigt den RPN-Stack vor und nach Aktivierung der Funktion COL+.





Funktion COL-

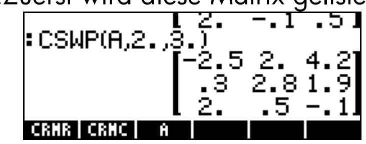
Die Funktion COL- verwendet als Argument eine Matrix und eine Integerzahl, welche die Position einer Spalte in der Matrix darstellt.Die Funktion gibt die ursprüngliche Matrix mit einer Spalte weniger, wieder, genauso wird auch die extrahierte Spalte als Vektor dargestellt. Hier ein Beispiel im ALG-Modus unter Verwendung der in A gespeicherten Matrix:



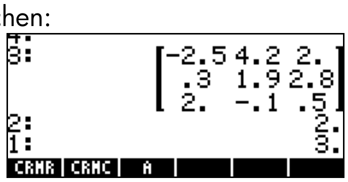
Im RPN-Modus laden Sie die Matrix erst in den Stack, dann geben Sie die Zahl, die eine Spalte der Matrix darstellt, vor Anwendung der Funktion COL- ein.Die nachfolgende Abbildung zeigt den RPN-Stack vor und nach Aktivierung der Funktion COL-.

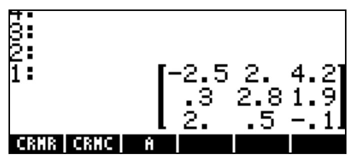
Funktion CSWP

Die Funktion CSWP (Column SWaP) verwendet als Argumente zwei Indizes, sagen wir i und j, (welche zwei unterschiedliche Spalten in der Matrix darstellen) und eine Matrix und erstellt daraus eine neue Matrix mit den Spalten i und j vertauscht.Das nachfolgende Beispiel, im ALG-Modus, zeigt die Anwendung dieser Funktion.Als Beispiel nehmen wir die in der Variablen A gespeicherte Matrix.Zuerst wird diese Matrix gelistet.



Im RPN-Modus können Sie mit der Funktion CSWP die Spalten einer Matrix in Stack Ebene 3, deren Indizes in Stack Ebene 1 und 2 aufgelistet sind, vertauschen.Als Beispiel, sehen Sie nachfolgend den RPN Stack vor und nach Anwendung der Funktion CSWP auf die Matrix A, um die Spalten 2 und 3 zu vertauschen:

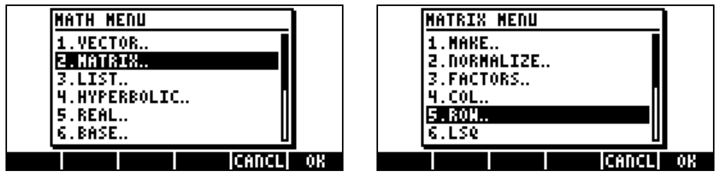




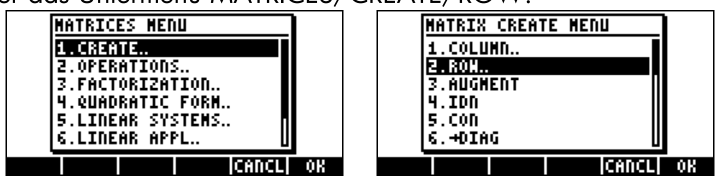
Wie Sie sehen können, wurden die Spalten, welche sich in Position 2 und 3 befunden haben, ausgetauscht. Der Austausch von Spalten und Zeilen (siehe unten) wird häufig bei der Lösung von linearen Gleichungen mit Matrizen verwendet.Eine genauere Beschreibung dieser Operationen erfolgt in einem späteren Kapitel.

Manipulation der Zeilen von Matrizen

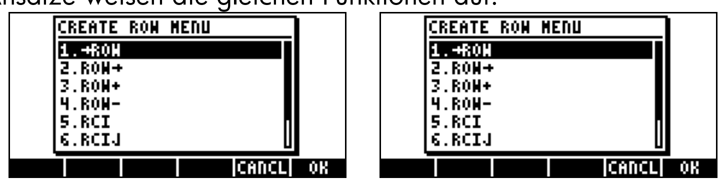
Der Rechner enthält ein Menü mit Funktionen zur Manipulation von Matrizen in deren Zeilen.Dieses Menü kann über MTH/MATRIX/ROW.. (⬅ MTH), wie in nachfolgender Abbildung, in Reihenfolge, gezeigt aufgerufen werden, während System Flag 117 auf CHOOSE Kästchen gesetzt wurde:



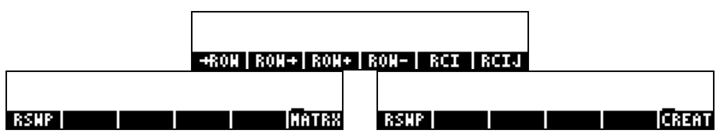
oder über das Untermenü MATRICES/CREATE/ROW:



Beide Ansätze weisen die gleichen Funktionen auf:



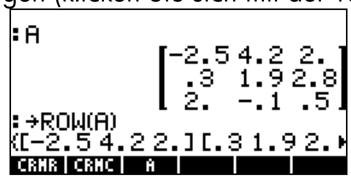
Ist das System Flag 117 auf SOFT Menüs gesetzt, kann das Menü ROW entweder über ← MTH MATRX MAKE ROW oder über ← MATRICES CREAT ROW aufgerufen werden. Beide Ansätze weisen die gleichen Funktionen auf:

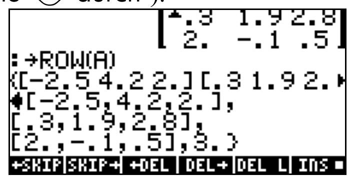


Die Anwendung dieser Funktionen wird nachfolgend dargestellt.

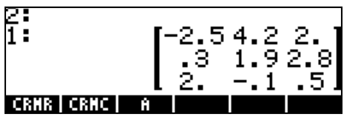
Funktion →ROW

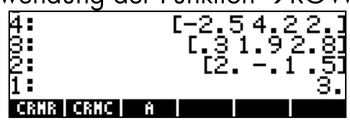
Die Funktion →ROW nimmt als Argument eine Matrix und zerlegt diese in Vektoren, entsprechend ihrer Zeilen.Nachfolgend wird eine Anwendung der Funktion →ROW im ALG-Modus gezeigt.Die verwendete Matrix wurde zu einen früheren Zeitpunkt bereits in der Variablen A gespeichert.Die Matrix ist in der linken Abbildung zu sehen: Die rechte Abbildung zeigt die in Zeilen zerlegte Matrix.Verwenden Sie den Zeileneditor, um das gesamte Ergebnis anzuzeigen (klicken Sie sich mit der Taste ▽ durch).





Im RPN-Modus müssen Sie die Matrix zuerst in den Stack laden, dann erst die Funktion →ROW, d.h., →ROW starten.Nachfolgende Abbildungen zeigen den RPN Stack vor und nach Anwendung der Funktion →ROW.



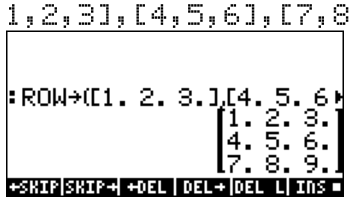


In diesem Ergebnis ist die erste Zeile, nach der Zerlegung, in der höchsten Stack Ebene, während in Stack Ebene 1 die Anzahl der Zeilen der ursprünglichen Matrix zu finden sind.Die Matrix bleibt bei der Zerlegung nicht erhalten, d.h. sie ist im Stack nicht mehr verfügbar.

Funktion ROW→

Die Funktion ROW→ hat die entgegensetzte Wirkung der Funktion →ROW, d.h. wenn Sie n Vektoren der gleiche Länge haben und die Zahl n, bildet die Funktion ROW→ eine Matrix, indem sie die eingegebenen Vektoren als Zeilen der Matrix darstellt.Hier ein Beispiel im ALG-Modus.Der verwendete Befehl war:

ROW→([1,2,3],[4,5,6],[7,8,9],3)

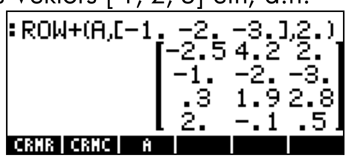


Geben Sie im RPN-Modus die n Vektoren in die Stack Ebenen n+1, n, n-1,…,2 ein und anschließend die Zahl n in Stack Ebene 1.So eingegeben, wird die Funktion ROW→ die Vektoren als Zeilen in die Matrix eingeben.

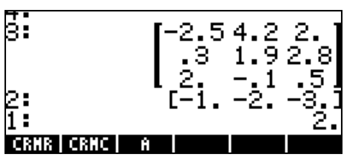
Die nachfolgende Abbildung zeigt den RPN-Stack vor und nach Aktivierung der Funktion ROW→.

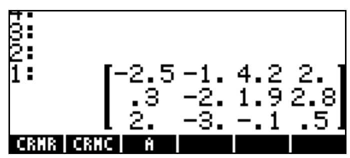
Funktion ROW+

Die Funktion ROW+ nimmt als Argument eine Matrix, einen Vektor der gleichen Länge wie die Anzahl der Zeilen in der Matrix und eine Integerzahl n, die die Position einer Zeile darstellt.Die Funktion ROW+ fügt den Vektor in Zeile n der Matrix ein.Als Beispiel setzen wir im ALG-Modus die zweite Zeile in Matrix A mit Hilfe des Vektors [-1,-2,-3] ein, d.h.



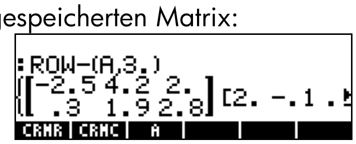
Im RPN-Modus geben wir zuerst die Matrix ein, dann den Vektor und die Nummer der Zeile, bevor wir die Funktion ROW+ anwenden.Die nachfolgende Abbildung zeigt den RPN-Stack vor und nach Aktivierung der Funktion ROW+.



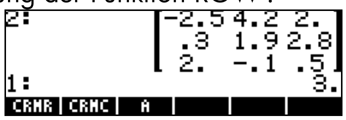


Function ROW-

Die Funktion ROW- verwendet als Argument eine Matrix und eine Integerzahl, welche die Position der Zeile in der Matrix darstellt.Die Funktion gibt die ursprüngliche Matrix mit einer Zeile weniger, wieder, genauso wird auch die extrahierte Zeile als Vektor dargestellt. Hier ein Beispiel im ALG-Modus unter Verwendung der in A gespeicherten Matrix:

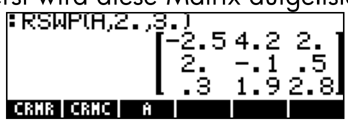


Im RPN-Modus laden Sie die Matrix erst in den Stack, dann geben Sie die Zahl, die eine Zeile der Matrix darstellt, vor Anwendung der Funktion ROW- ein.Die nachfolgende Abbildung zeigt den RPN-Stack vor und nach Aktivierung der Funktion ROW-.

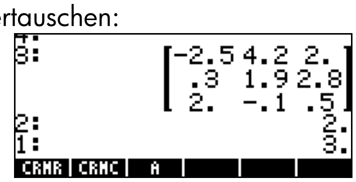


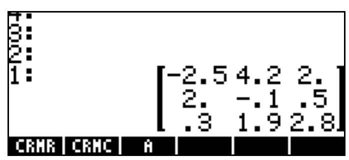
Funktion RSWP

Die Funktion RSWP (Row SWaP) verwendet als Argument zwei Indizes, sagen wir i und j, (welche zwei unterschiedliche Zeilen in der Matrix darstellen) und eine Matrix und erstellt daraus eine neue Matrix mit den Zeilen i und j vertauscht.Das nachfolgende Beispiel, im ALG-Modus, zeigt die Anwendung dieser Funktion.Als Beispiel nehmen wir die in der Variablen A gespeicherte Matrix.Zuerst wird diese Matrix aufgelistet.



Im RPN-Modus können Sie mit der Funktion RSWP die Zeilen einer Matrix angezeigt in Stack Ebene 3, deren Indizes in Stack Ebene 1 und 2 aufgelistet sind, vertauschen.Als Beispiel, sehen Sie nachfolgend den RPN Stack vor und nach Anwendung der Funktion RSWP auf die Matrix A, um die Zeilen 2 und 3 zu vertauschen:

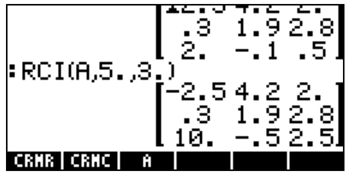




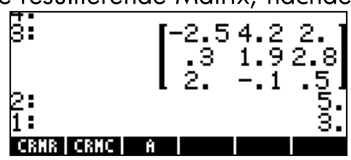
Wie Sie sehen können, wurden die Spalten, welche sich in Position 2 und 3 befunden haben, ausgetauscht.

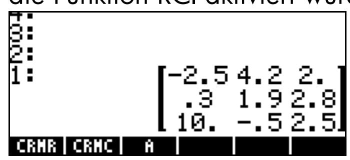
Funktion RCI

Die Funktion RCI steht für Multiplikation der Zeile I mit einem konstanten Wert und Ersetzen der so entstandenen Zeile an der gleichen Stelle wie die vorhergehende. Das nachfolgende Beispiel, im ALG-Modus, nimmt die in der Variablen A gespeicherte Matrix und multipliziert den konstanten Wert 5 mit der Zeile Nr. 3, und ersetzt diese Zeile mit dem Produkt der Multiplikation.



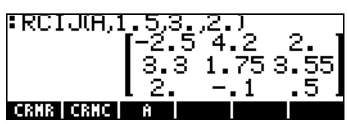
Die gleiche Übung wird in nachfolgender Abbildung im RPN-Modus angezeigt. Die linke Abbildung zeigt die Erstellung der Matrix, den Faktor und die Anzahl der Zeilen in Stack Ebene 3, 2 und 1.Die rechte Abbildung zeigt die resultierende Matrix, nachdem die Funktion RCI aktiviert wurde.





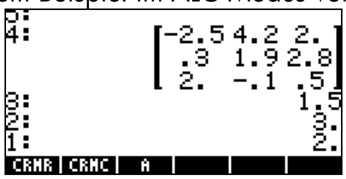
Funktion RCIJ

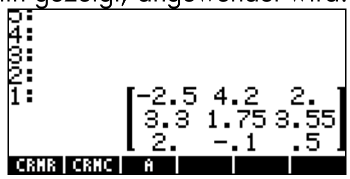
Die Funktion RCIJ steht für “nehme Zeile I und multipliziere diese mit der Konstanten C, setze dann die multiplizierte Reihe in Reihe J und ersetze die Reihe J mit der errechneten Summe".Diese Art von Operation wird häufig in der Gaußschen oder der Gauß-Jordan Elimination verwendet (weitere Einzelheiten zu dieser Prozedur finden Sie in einem späteren Kapitel).Die Argumente der Funktion sind: (1) die Matrix, (2) der konstante Wert, (3) die Zeile die mit der Konstanten in (2) multipliziert werden soll und (4) die Zeile, die mit dem Ergebnis, wie oben beschrieben, ersetzt werden soll. Nehmen wir z.B. die in A gespeicherte Matrix, multiplizieren wir die dritte Spalte mit 1,5 und addieren diese zu Spalte 2.Nachfolgendes Beispiel wird im ALG-Modus ausgeführt:



Im RPN-Modus, geben Sie zuerst die Matrix, gefolgt von der Konstanten, ein dann die Zeile, die mit der Konstanten multipliziert werden soll und schließlich dann die Zeile, die ersetzt werden soll.Die nachfolgende Abbildung zeigt den

RPN Stack vor und nachdem die Funktion RCIJ, unter denselben Bedingungen wie in dem Beispiel im ALG-Modus vorhin gezeigt, angewendet wird:





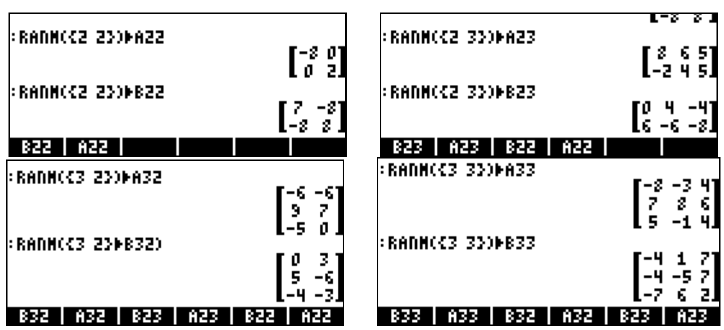
Kapitel 11
Matrix-Operationen und lineare Algebra

In Kapitel 10 führten wir das Konzept der Matrix ein und stellten mehrere Funktionen zum Eingeben, Erstellen und Bearbeiten von Matrizen vor. In diesem Kapitel präsentieren wir Beispiele für Matrix-Operationen und -Anwendungen in Bezug auf Probleme der linearen Algebra.

Operationen mit Matrizen

Matrizen können wie andere mathematische Objekte addiert und subtrahiert werden. Sie können mit einem Skalar oder auch miteinander multipliziert werden. Eine für Anwendungen der linearen Algebra wichtige Operation ist die Bildung der Inversen einer Matrix. Diese Operationen werden nun dargestellt.

Zur Veranschaulichung der Operationen erstellen wir mehrere Matrizen, die wir in Variablen speichern. Die generischen Namen der Matrizen lauten A*ij* und B*ij*, wobei *i* die Anzahl der Zeilen und *j* die Anzahl der Spalten der Matrizen darstellen. Die verwendeten Matrizen werden mithilfe der Funktion RANM (Zufallsmatrizen) erzeugt. Wenn Sie diese Übung mit dem Taschenrechner durchführen, erhalten Sie andere als die hier aufgeführten Matrizen, sofern Sie sie nicht genau wie unten dargestellt im Taschenrechner speichern. Unten sind die im ALG-Modus erzeugten Matrizen A22, B22, A23, B23, A32, B32, A33 und B33 dargestellt:

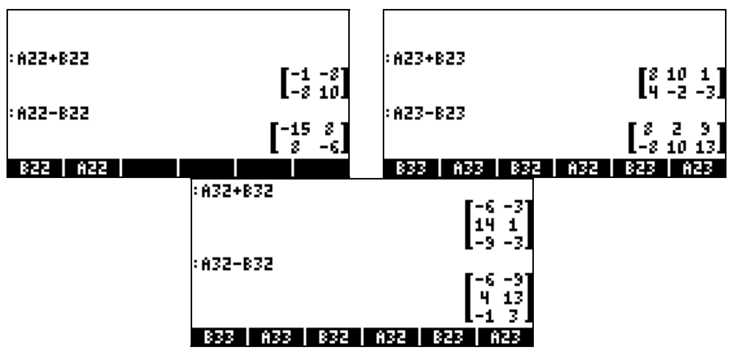


Im RPN-Modus lauten die Schritte wie folgt:

```
{2,2} ENTER RANM 'A22' STO▶    {2,2} ENTER RANM 'B22' STO▶
{2,3} ENTER RANM 'A23' STO▶    {2,3} ENTER RANM 'B23' STO▶
{3,2} ENTER RANM 'A32' STO▶    {3,2} ENTER RANM 'B32' STO▶
{3,3} ENTER RANM 'A33' STO▶    {2,2} ENTER RANM 'B33' STO▶
```

Addition und Subtraktion

Gegeben seien zwei Matrizen **A** = $[a_{ij}]_{m\times n}$ und **B** = $[b_{ij}]_{m\times n}$. Diese beiden Matrizen können nur addiert bzw. subtrahiert werden, wenn die Anzahl ihrer Zeilen und Spalten übereinstimmt. Die resultierende Matrix **C** = **A** ± **B** = $[c_{ij}]_{m\times n}$ besitzt die Elemente $c_{ij} = a_{ij} \pm b_{ij}$. Nachfolgend finden Sie einige Beispiele im ALG Modus, unter Verwendung der oben gespeicherten Matrizen (z.B. A22 + B22)



Im RPN-Modus lauten die Schritte wie folgt:

```
A22 ENTER B22 ENTER +     A22 ENTER B22 ENTER −
A23 ENTER B23 ENTER +     A23 ENTER B23 ENTER −
A32 ENTER B32 ENTER +     A32 ENTER B32 ENTER −
A33 ENTER B33 ENTER +     A33 ENTER B33 ENTER −
```

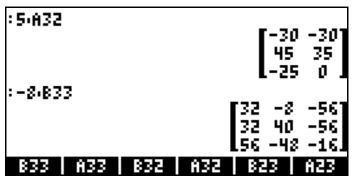
Wie hier veranschaulicht, ist die Umwandlung der Beispiele vom ALG-Modus in den RPN-Modus einfach. Die übrigen Beispiele für Matrix-Operationen werden ausschließlich im ALG-Modus durchgeführt.

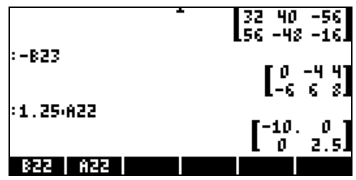
Multiplikation

Es sind mehrere Multiplikationsoperationen mit Matrizen möglich. Diese werden im Folgenden beschrieben.

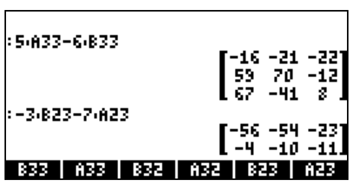
Multiplikation mit einem Skalar

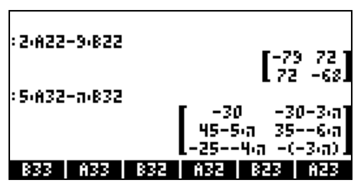
Durch Multiplikation der Matrix **A** = $[a_{ij}]_{m\times n}$ mit einem Skalar ergibt sich die Matrix **C** = k**A** = $[c_{ij}]_{m\times n}$ = $[ka_{ij}]_{m\times n}$. Eine negative Matrix wird durch die Operation -**A** =(-1)**A** = $[-a_{ij}]_{m\times n}$ definiert. Unten sind einige Beispiele für die Multiplikation einer Matrix mit einem Skalar dargestellt.



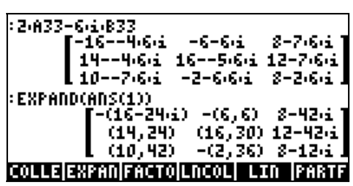


Durch die Kombination von Addition und Subtraktion mit der Skalarmultiplikation können wir lineare Kombinationen von Matrizen derselben Dimension bilden, z. B.



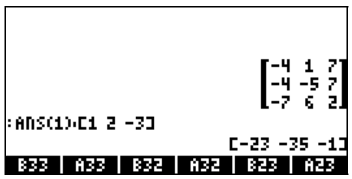


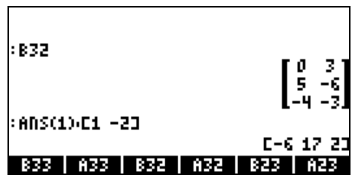
In einer Linearkombination von Matrizen können wir eine Matrix mit einer imaginären Zahl multiplizieren, um eine Matrix komplexer Zahlen zu erhalten, z. B.



Matrix-Vektor-Multiplikation

Die Matrix-Vektor-Multiplikation ist nur dann möglich, wenn die Anzahl der Matrix-Spalten mit der Länge des Vektors übereinstimmt. Diese Operation erfolgt nach den im nächsten Abschnitt dargestellten Regeln der Matrix-Multiplikation. Es folgen mehrere Beispiele für die Matrix-Vektor-Multiplikation:





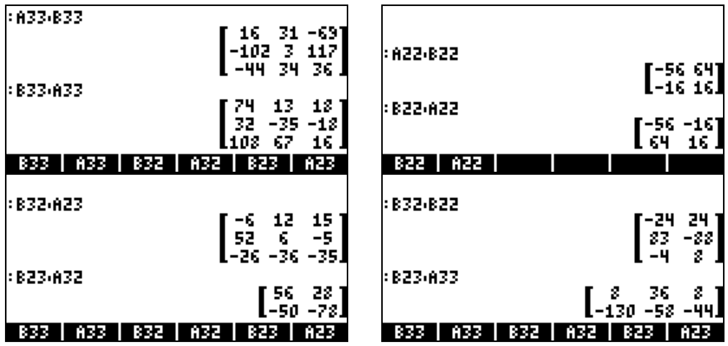
Die Vektor-Matrix-Multiplikation ist hingegen nicht definiert. Diese Multiplikation kann jedoch als spezieller Fall der im Folgenden definierten Matrix-Multiplikation ausgeführt werden.

Matrix-Multiplikation

Die Matrix-Multiplikation ist durch $\mathbf{C}_{m\times n} = \mathbf{A}_{m\times p}\cdot\mathbf{B}_{p\times n}$ definiert, wobei $\mathbf{A} = [a_{ij}]_{m\times p}$, $\mathbf{B} = [b_{ij}]_{p\times n}$ und $\mathbf{C} = [c_{ij}]_{m\times n}$. Beachten Sie, dass die Matrix-Multiplikation nur dann möglich ist, wenn die Anzahl der Spalten im ersten Operanden gleich der Anzahl der Zeilen im zweiten Operanden ist. Die allgemeine Größe des Produkts, c_{ij}, ist definiert als

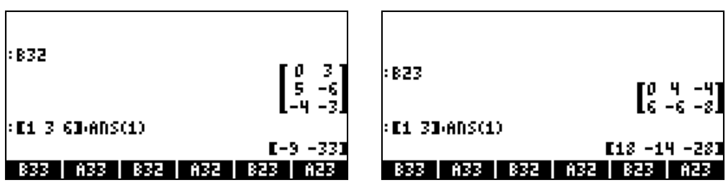
$$c_{ij} = \sum_{k=1}^{p} a_{ik} \cdot b_{kj},\ for\ i = 1,2,\ldots,m;\ j = 1,2,\ldots,n.$$

Dies bedeutet, dass das Element in der i-ten Zeile und j-ten Spalte des Produkts, **C**, dadurch gebildet wird, dass die einzelnen Größen der i-ten Zeile von **A** mit den einzelnen Größen der j-ten Spalte von **B** multipliziert und die Produkte addiert werden. Die Matrix-Multiplikation ist nicht kommutativ, d. h., allgemein gilt: $\mathbf{A}\cdot\mathbf{B} \neq \mathbf{B}\cdot\mathbf{A}$. Darüber hinaus kann sogar eine der Multiplikationen nicht vorhanden sein. In den folgenden Bildschirmabbildungen werden die Ergebnisse der Multiplikationen der zuvor gespeicherten Matrizen dargestellt:



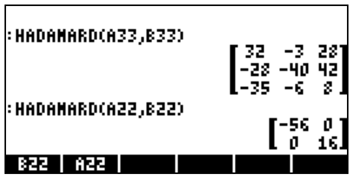
Die im vorherigen Abschnitt vorgestellte Matrix-Vektor-Multiplikation kann als Produkt einer m×n-Matrix mit einer n×1-Matrix (d. h. einem Spaltenvektor) gedacht werden, der eine m×1-Matrix (also einen anderen Vektor) ergibt. Überprüfen Sie die im vorherigen Abschnitt dargestellten Beispiele, um diese Aussage zu verifizieren. Für den Zweck der Matrixmultiplikation sind daher die in Kapitel 9 definierten Vektoren im Grunde genommen Spaltenvektoren.

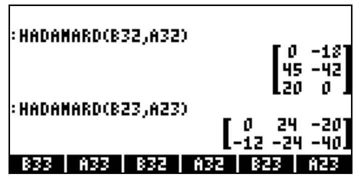
Das Produkt eines Vektors mit einer Matrix kann gebildet werden, wenn der Vektor ein Zeilenvektor ist, d. h. eine 1×m-Matrix, die bei der Multiplikation mit einer m×n-Matrix eine 1×n-Matrix ergibt. Damit der Taschenrechner einen Zeilenvektor erkennen kann, müssen Sie diesen in Klammern eingeben, z. B.



Multiplikation der einzelnen Größen nacheinander

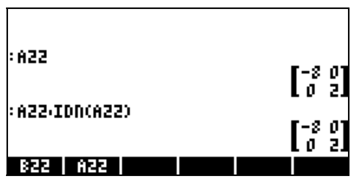
Die Multiplikation der einzelnen Größen zweier Matrizen derselben Dimension nacheinander ist durch Verwendung der Funktion HADAMARD möglich. Das Ergebnis ist natürlich eine weitere Matrix derselben Dimension. Diese Funktion ist über den Funktionskatalog (↱ _CAT_) oder über das Untermenü MATRICES/OPERATIONS (↰ _MATRICES_) verfügbar. Im Folgenden werden Anwendungen der Funktion HADAMARD vorgestellt:





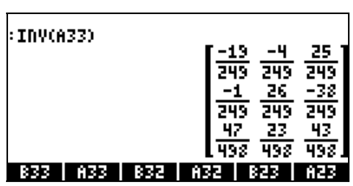
Die Einheitsmatrix

In Kapitel 9 wird die Einheitsmatrix als Matrix **I** = $[\delta_{ij}]_{n\times n}$ vorgestellt, wobei δ_{ij} die Kronecker-Deltafunktion darstellt. Einheitsmatrizen können durch Verwendung der in Kapitel 9 beschriebenen Funktion IDN erzeugt werden. Für die Einheitsmatrix gilt **A**·**I** = **I**·**A** = **A**. Zur Überprüfung dieser Eigenschaft stellen wir die folgenden Beispiele dar und verwenden hierfür die bereits gespeicherten Matrizen:

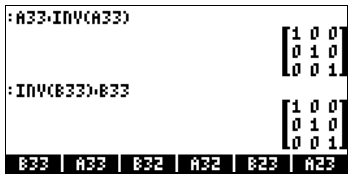


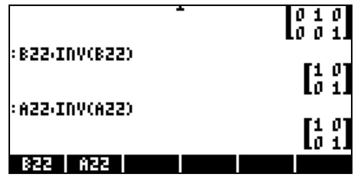
Die inverse Matrix

Die Inverse einer quadratischen Matrix ist die Matrix $\mathbf{A}^{-1}$, so dass $\mathbf{A}\cdot\mathbf{A}^{-1} = \mathbf{A}^{-1}\cdot\mathbf{A} = \mathbf{I}$, wobei **I** die Einheitsmatrix mit derselben Dimension wie **A** ist. Mit dem Taschenrechner erhalten Sie die Inverse einer Matrix durch die Umkehrfunktion INV (d. h mit der 1/x-Taste). Unten sind Beispiele für die Inverse einiger zuvor gespeicherter Matrizen dargestellt:



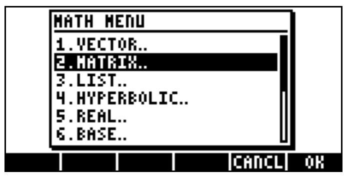
Zur Überprüfung der Eigenschaften der inversen Matrix stellen wir die folgenden Multiplikationen dar:





Beschreiben einer Matrix (Das Matrixmenü NORM)

Das Matrixmenü NORM (NORMALIZE) wird mit der Tastenkombination MTH aufgerufen (der Systemflag 117 ist auf die Felder CHOOSE gesetzt):



Das Menü enthält die folgenden Funktionen:

Diese Funktionen werden im Folgenden beschrieben. Da viele dieser Funktionen Konzepte der Matrixtheorie, z. B. Singulärwerte, Rang usw., verwenden, enthalten die Beschreibungen der Funktionen kurze Darstellungen dieser Konzepte.

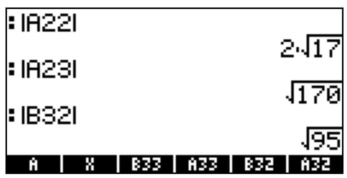
Funktion ABS

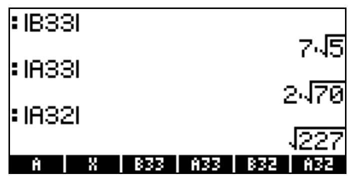
Mit der Funktion ABS wird die Frobenius-Norm einer Matrix berechnet. Für eine Matrix **A** = $[a_{ij}]_{m\times n}$ wird die Frobenius-Norm einer Matrix definiert als

$$\|A\|_F = \sqrt{\sum_{i=1}^{n}\sum_{j=1}^{m} a_{ij}^2}$$

Wenn es sich bei der betreffenden Matrix um einen Zeilen- oder Spaltenvektor handelt, ist die Frobenius-Norm $||\mathbf{A}||_F$ einfach der Betrag des Vektors. Die Funktion ABS kann direkt über die Tastenkombination ABS aufgerufen werden.

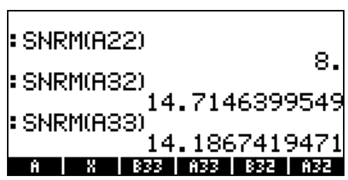
Führen Sie im ALG-Modus die folgenden Übungen durch (mit den zuvor für Matrix-Operationen gespeicherten Matrizen):





Funktion SNRM

Mit der Funktion SNRM wird die Spektralnorm einer Matrix berechnet, die als der größte Singulärwert der Matrix definiert ist und auch als euklidische Norm der Matrix bezeichnet wird. Beispiel:



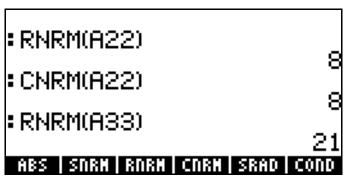
Singulärwertzerlegung

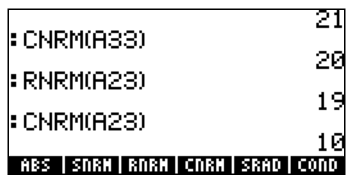
Zum Verständnis der Funktion SNRM müssen wir das Konzept der Matrixzerlegung erläutern. Die Matrixzerlegung umfasst im Wesentlichen die Bestimmung von mindestens zwei Matrizen, die die ursprüngliche Matrix ergeben, wenn sie in einer bestimmten Reihenfolge (eventuell mit Matrixinversion oder -transposition) multipliziert werden. Bei der Singulärwertzerlegung (SVD) wird eine rechteckige Matrix $\mathbf{A}_{m\times n}$ als $\mathbf{A}_{m\times n} = \mathbf{U}_{m\times m} \cdot \mathbf{S}_{m\times n} \cdot \mathbf{V}^T_{n\times n}$ beschrieben, wobei es sich bei **U** und **V** um Orthogonalmatrizen und bei **S** um eine Diagonalmatrix handelt. Die diagonalen Elemente von **S** werden als Singulärwerte von A bezeichnet und sind in der Regel so angeordnet, dass für i = 1, 2, …, n-1 gilt, dass $s_i \geq s_{i+1}$. Die Spalten [$\mathbf{u}_j$] von **U** und [$\mathbf{v}_j$] von **V** sind die entsprechenden Singulärvektoren. (Für Orthogonalmatrizen gilt: $\mathbf{U} \cdot \mathbf{U}^T = \mathbf{I}$. Eine Diagonalmatrix besitzt nur entlang ihrer Hauptdiagonalen Elemente ungleich Null.)

Der Rang einer Matrix kann anhand ihrer SVD bestimmt werden, indem die Anzahl der nichtsingulären Wert gezählt wird. Der nächste Abschnitt enthält Beispiele für die Singulärwertzerlegung.

Funktionen RNRM und CNRM

Die Funktion RNRM gibt die Zeilennorm einer Matrix und die Funktion CNRM die Spaltennorm einer Matrix zurück. Beispiele:

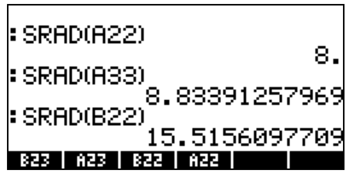




Zeilennnorm und Spaltennorm einer Matrix
Die Zeilennorm einer Matrix wird berechnet, indem die Summen der absoluten Werte aller Elemente in jeder Zeile gebildet werden und anschließend der Höchstwert dieser Summen ausgewählt wird. Die Spaltennorm einer Matrix wird berechnet, indem die Summen der absoluten Werte aller Elemente in jeder Spalte gebildet werden und anschließend der Höchstwert dieser Summen ausgewählt wird.

Funktion SRAD

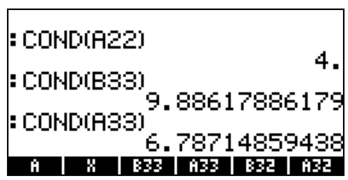
Mit der Funktion SRAD wird der Spektralradius einer Matrix bestimmt. Dieser ist als der größte absolute Wert der Eigenwerte einer Matrix definiert. Beispiel:



Definition der Eigenwerte und Eigenvektore einer Matrix
Die Eigenwerte einer quadratischen Matrix sind das Ergebnis der Matrixgleichung $\mathbf{A}\cdot\mathbf{x} = \lambda\cdot\mathbf{x}$. Die der Gleichung entsprechenden Werte von λ werden als Eigenwerte der Matrix **A** bezeichnet. Die für jeden Wert von λ aus der Gleichung resultierenden Werte von x werden als Eigenvektoren der Matrix bezeichnet. Weitere Informationen über das Berechnen von Eigenwerten und Eigenvektoren erhalten Sie im nächsten Kapitel.

Funktion COND

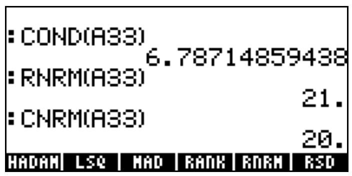
Mit der Funktion COND wird die Konditionszahl einer Matrix bestimmt. Beispiele:

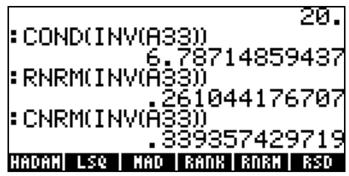


Konditionszahl einer Matrix
Die Konditionszahl einer quadratischen nichtsingulären Matrix ist als das Produkt der Matrixnorm und der Norm ihrer Inversen definiert, d. h. $\mathrm{cond}(\mathbf{A}) = ||\mathbf{A}|| \times ||\mathbf{A}^{-1}||$. Wir wählen als Matrixnorm $||\mathbf{A}||$ den Höchstwert ihrer Zeilennorm (RNRM) und ihrer Spaltennorm (CNRM), während als Norm ihrer Inversen $||\mathbf{A}^{-1}||$ der Mindestwert ihrer Zeilennorm und Spaltennorm gewählt wird. Somit gilt $||\mathbf{A}|| = \max(\mathrm{RNRM}(\mathbf{A}),\mathrm{CNRM}(\mathbf{A}))$ und $||\mathbf{A}^{-1}|| = \min(\mathrm{RNRM}(\mathbf{A}^{-1}), \mathrm{CNRM}(\mathbf{A}^{-1}))$.

Die Konditionszahl einer singulären Matrix ist Unendlich. Die Konditionszahl einer nichtsingulären Matrix bestimmt, wie weit die Matrix von der Singularität entfernt ist. Je größer die Konditionszahl, desto näher befindet sich die Matrix an der Singularität. (Eine singuläre Matrix ist eine Matrix, für die keine inverse Matrix vorhanden ist.)

Führen Sie für Matrix A33 folgende Übung zur Matrixkonditionszahl durch. Die Konditionszahl lautet COND(A33). Zeilennorm und Spaltennorm für A33 werden auf der linken Seite angezeigt. Die entsprechenden Zahlen für die inverse Matrix INV(A33) werden auf der linken Seite angezeigt:

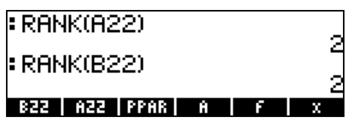




Da RNRM(A33) > CNRM(A33), ist ||A33|| = RNRM(A33) = 21. Da außerdem CNRM(INV(A33)) < RNRM(INV(A33)), ist ||INV(A33)|| = CNRM(INV(A33)) = 0.261044... Die Konditionszahl wird somit als CNRM(A33)*CNRM(INV(A33)) = COND(A33) = 6.7871485… berechnet.

Funktion RANK

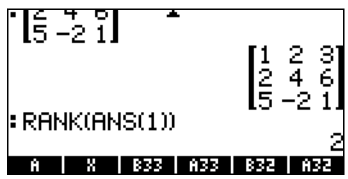
Mit der Funktion RANK wird der Rang einer quadratischen Matrix bestimmt. Testen Sie folgende Beispiele:



Rang einer Matrix

Der Rang einer quadratischen Matrix ist die maximale Anzahl linear unabhängiger Zeilen oder Spalten der Matrix. Angenommen, Sie erstellen eine quadratische Matrix $\mathbf{A}_{n\times n}$ als $\mathbf{A} = [\mathbf{c}_1\ \mathbf{c}_2 \ldots \mathbf{c}_n]$, wobei $\mathbf{c}_i$ (i = 1, 2, …, n) Vektoren sind, die die Spalten von Matrix **A** darstellen. Wenn dann eine dieser Spalten, z. B. $\mathbf{c}_k$, als $\mathbf{c}_k = \sum_{j\neq k,\, j\in\{1,2,\ldots,n\}} d_j \cdot \mathbf{c}_j$, beschrieben werden kann, wobei die Werte d_j konstant sind, ist $\mathbf{c}_k$ von den in der Summe enthaltenen Spalten linear unabhängig. (Beachten Sie, dass die Werte von j jeden Wert in der Menge {1, 2, …, n} in jeder beliebigen Kombination enthalten, solange $j \neq k$.) Wenn der obige Ausdruck für keinen der Spaltenvektoren gebildet werden kann, sind alle Spalten linear unabhängig. Eine vergleichbare Definition der linearen Unabhängigkeit von Zeilen kann entwickelt werden, indem die Matrix als eine Spalte von Zeilenvektoren dargestellt wird. Wenn daher rank(**A**) = n, besitzt die Matrix eine Inverse und ist eine nichtsinguläre Matrix. Wenn hingegen rank(**A**) < n, ist die Matrix singulär und keine Inverse vorhanden.

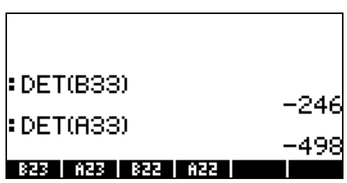
Bestimmen Sie beispielsweise den Rang der folgenden Matrix:

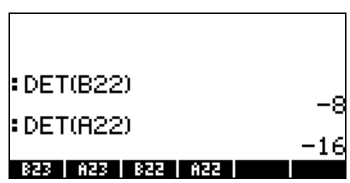


Der Rang ist 2. Der Grund hierfür ist, dass die zweite Zeile [2,4,6] gleich dem Produkt der ersten Zeile [1,2,3] mit 2 ist. Somit ist Zeile zwei von Zeile 1 linear abhängig und die maximale Anzahl linear unabhängiger Zeilen ist 2. Sie können überprüfen, ob die maximale Anzahl linear unabhängiger Zeilen 3 ist. Der Rang, die maximale Anzahl linear unabhängiger Zeilen oder Zeilen, ist in diesem Fall 2.

Funktion DET

Mit der Funktion DET wird die Determinante einer quadratischen Matrix berechnet. Beispiel:



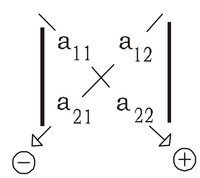


Determinante einer Matrix

Die Determinanten einer 2x2- und einer 3x3-Matrix werden durch dieselbe Anordnung der Elemente der Matrizen dargestellt, jedoch zwischen vertikalen Linien, also

$$\begin{vmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{vmatrix}, \qquad \begin{vmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{vmatrix}$$

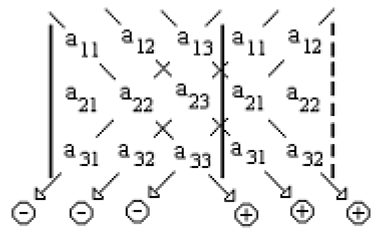
Eine 2×2-Determinante wird berechnet, indem die Elemente auf ihrer Diagonalen multipliziert und diese Produkte mit positivem bzw. negativem Vorzeichen addiert werden, wie im Diagramm unten dargestellt.



Die 2×2-Determinante lautet daher

$$\begin{vmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{vmatrix} = a_{11} \cdot a_{22} - a_{12} \cdot a_{21}$$

Eine 3×3-Determinante wird berechnet, indem die Determinante erweitert wird. Dies geschieht, indem die ersten beiden Spalten der Determinante kopiert und rechts von Spalte 3 eingefügt werden, wie im Diagramm unten dargestellt. Im Diagramm sind auch Elemente dargestellt, die auf die gleiche Weise wie zuvor bei einer 2×2-Determinante multipliziert werden, wobei das Produkt das entsprechende Vorzeichen erhält. Nach der Multiplikation werden die Ergebnisse addiert, um die Determinante zu erhalten.



Determinanten für quadratische Matrizen höherer Ordnung können mithilfe von Determinanten niedrigerer Ordnung, die als Kofaktor bezeichnet werden, berechnet werden. Hierbei wird die Determinante einer n×n-Matrix (auch als n×n-Determinante bezeichnet) zu einer Summe der Kofaktoren „erweitert", bei denen es sich um (n-1)×(n-1) Determinanten handelt, multipliziert mit den Elementen einer einzelnen Zeile oder Spalte, wobei die Vorzeichen abwechselnd positiv und negativ sind. Diese „Erweiterung" wird mit den Kofaktoren der Ordnung (n-2)×(n-2) auf die nächste (niedrigere) Ebene übertragen usw., bis nur noch eine umfangreiche Summe von 2×2-Determinanten vorhanden ist. Die 2×2-Determinanten werden dann mit der oben dargestellten Methode berechnet.

Die Berechnung einer Determinante durch Entwicklung der Kofaktoren ist insofern sehr ineffizient, als sie Operationen umfasst, deren Anzahl mit der Größe der Determinante sehr schnell zunimmt. Eine effizientere und bei numerischen Anwendungen bevorzugte Methode besteht darin, das Ergebnis einer Gauß-Elimination zu verwenden. Die Gauß-Elimination wird zum Lösen linearer Gleichungssysteme verwendet. Diese Methode wird weiter unten in diesem Kapitel dargestellt.

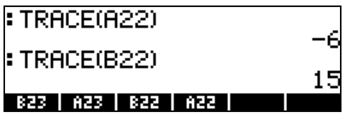
Die Determinante einer Matrix **A** wird als det(**A**) dargestellt. Die Determinante einer singulären Matrix ist gleich Null.

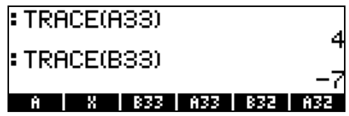
Funktion TRACE

Mit der Funktion TRACE wird die Spur einer quadratischen Matrix berechnet, die als die Summe der Elemente ihrer Hauptdiagonalen definiert ist oder als

$$tr(\mathbf{A}) = \sum_{i=1}^{n} a_{ii} \ .$$

Beispiele:





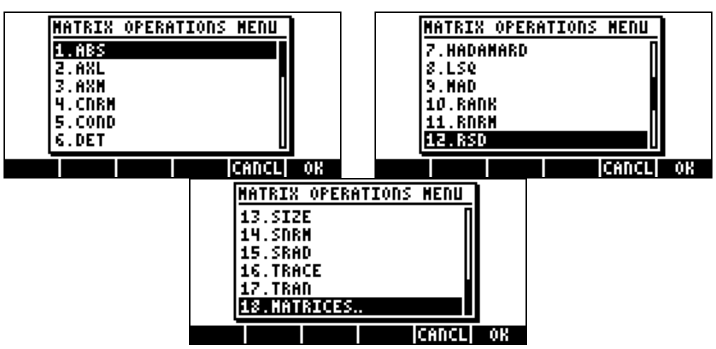
Funktion TRAN

Die Funktion TRAN gibt die Transponierte einer reellen Matrix oder die konjugierte Transponierte einer komplexen Matrix zurück. TRAN ist mit TRN äquivalent. Die Funktion TRN wurde in Kapitel 10 erläutert.

Weitere Matrix-Operationen (Das Matrix-Menü OPER)

Das Matrixmenü OPER (OPERATIONS) wird mit der Tastenkombination MATRICES aufgerufen (der Systemflag 117 ist auf die Felder CHOOSE gesetzt):

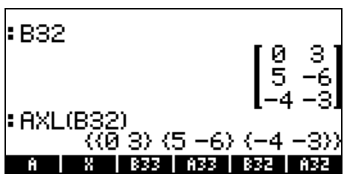
Das Menü OPERATIONS enthält die folgenden Funktionen:

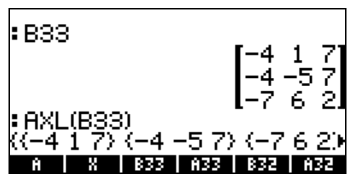


Die Funktionen ABS, CNRM, COND, DET, RANK, RNRM, SNRM, TRACE und TRAN sind auch im Menü MTH/MATRIX/NORM (das Thema des vorherigen Abschnitts) verfügbar. Die Funktion SIZE wurde in Kapitel 10 dargestellt. Die Funktion HADAMARD wurde bereits im Zusammenhang mit der Matrix-Multiplikation vorgestellt. Die Funktionen LSQ, MAD und RSD werden bei der Lösung linearer Gleichungssysteme verwendet und in einem späteren Abschnitt dieses Kapitels dargestellt. In diesem Abschnitt werden nur die Funktionen AXL und AXM erläutert.

Funktion AXL

Mit der Funktion AXL wird ein Feld (eine Matrix) in eine Liste umgewandelt und umgekehrt. Beispiele:

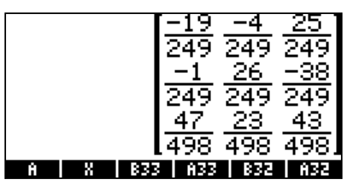


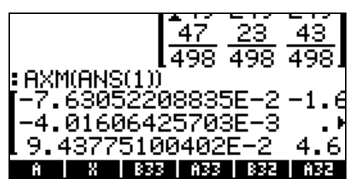


Hinweis: Die letzte Operation ist mit der Operation des in Kapitel 10 dargestellten Programms CRMR vergleichbar.

Funktion AXM

Mit der Funktion AXM wird ein Feld mit ganzen Zahlen oder Brüchen in die entsprechende Dezimal- oder Näherungsform umgewandelt. Beispiel:





Funktion LCXM

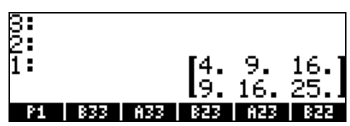
Mit der Funktion LCXM können Matrizen erzeugt werden, für die gilt, dass das Element aij eine Funktion von i und j ist. Die Eingangswerte dieser Funktion sind zwei Ganzzahlen n und m, die die Anzahl der Zeilen und Spalten der zu erzeugenden Matrix darstellen, und ein Programm mit den Eingangswerten i und j. Die Zahlen n und m sowie das Programm belegen jeweils Ebene 3, 2 und 1 des Stacks. Die Funktion LCXM kann über den Befehlskatalog ↱ _CAT_ aufgerufen werden.

Um beispielsweise eine 2´3-Matrix zu erzeugen, deren Elemente durch a_{ij} = $(i+j)^2$ gegeben sind, speichern Sie zunächst im RPN-Modus das folgende Programm in der Variablen P1. Bevor Sie STO▸ drücken, wird der RPN-Stack wie folgt angezeigt:

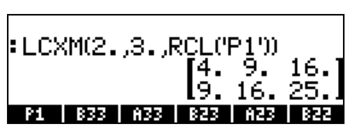
Die Ausführung der Funktion LCXM erfordert in diesem Fall, dass Sie Folgendes eingeben:

2 ENTER 3 ENTER ↱ LCXM ENTER

In der folgenden Abbildung ist der RPN-Stack vor und nach Anwendung der Funktion LCXM dargestellt:



Im ALG-Modus erhalten Sie dieses Beispiel durch folgende Eingabe:



Das Programm P1 muss jedoch im RPN-Modus erstellt und gespeichert worden sein.

Lösung linearer Gleichungssysteme

Ein System von *n* linearen Gleichungen mit *m* Variablen kann folgendermaßen beschrieben werden:

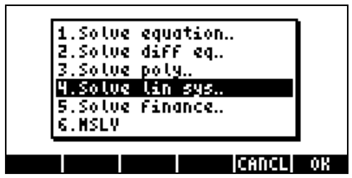
$$\begin{array}{l}
a_{11}\cdot x_1 + a_{12}\cdot x_2 + a_{13}\cdot x_3 + \ldots + a_{1,m-1}\cdot x_{m-1} + a_{1,m}\cdot x_m = b_1 \\
a_{21}\cdot x_1 + a_{22}\cdot x_2 + a_{23}\cdot x_3 + \ldots + a_{2,m-1}\cdot x_{m-1} + a_{2,m}\cdot x_m = b_2 \\
a_{31}\cdot x_1 + a_{32}\cdot x_2 + a_{33}\cdot x_3 + \ldots + a_{3,m-1}\cdot x_{m-1} + a_{3,m}\cdot x_m = b_3 \\
\quad . \qquad . \qquad . \qquad \ldots \qquad . \qquad . \qquad . \\
\quad . \qquad . \qquad . \qquad \ldots \qquad . \qquad . \qquad . \\
a_{n-1,1}\cdot x_1 + a_{n-1,2}\cdot x_2 + a_{n-1,3}\cdot x_3 + \ldots + a_{n-1,m-1}\cdot x_{m-1} + a_{n-1,m}\cdot x_m = b_{n-1} \\
a_{n1}\cdot x_1 + a_{n2}\cdot x_2 + a_{n3}\cdot x_3 + \ldots + a_{n,m-1}\cdot x_{m-1} + a_{n,m}\cdot x_m = b_n.
\end{array}$$

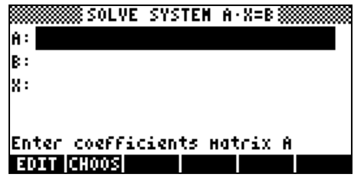
Dieses lineare Gleichungssystem kann als Matrixgleichung $\mathbf{A}_{n\times m}\cdot\mathbf{x}_{m\times 1} = \mathbf{b}_{n\times 1}$ beschrieben werden, wenn wir folgende Matrix und Vektoren definieren:

$$A = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1m} \\ a_{21} & a_{22} & \cdots & a_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nm} \end{bmatrix}_{n\times m}, \quad x = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_m \end{bmatrix}_{m\times 1}, \quad b = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{bmatrix}_{n\times 1}$$

Verwenden der numerischen Lösung für lineare Gleichungssysteme

Lineare Gleichungssysteme können mit dem Taschenrechner auf viele Arten gelöst werden. Eine Möglichkeit besteht in der Verwendung der numerischen Lösung ⇨ NUM.SLV . Wählen Sie im unten (links) angezeigten Fenster der numerischen Lösung die Option *4. Solve lin sys* aus und drücken Sie OK. Anschließend wird folgende Eingabemaske angezeigt (rechts):





Um das lineare Gleichungssystem **A**·**x** = **b** zu lösen, geben Sie die Matrix **A** im Format [[a_{11}, $a_{12,}$ …], … [….]] in das Feld A: ein. Geben Sie außerdem den Vektor **b** in das Feld B: ein. Wenn das Feld X: markiert ist, drücken Sie [SOLVE]. Ist eine Lösung verfügbar, wird im Feld X: der Lösungsvektor **x** angezeigt. Die Lösung wird außerdem in Ebene 1 des Stacks kopiert. Es folgen einige Beispiele.

Ein quadratisches Gleichungssystem

Das lineare Gleichungssystem

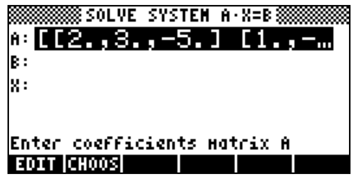
$$2x_1 + 3x_2 - 5x_3 = 13,$$
$$x_1 - 3x_2 + 8x_3 = -13,$$
$$2x_1 - 2x_2 + 4x_3 = -6$$

kann als Matrixgleichung **A**·**x** = **b** beschrieben werden, wenn

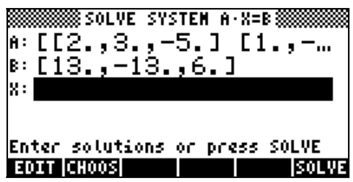
$$\mathbf{A} = \begin{bmatrix} 2 & 3 & -5 \\ 1 & -3 & 8 \\ 2 & -2 & 4 \end{bmatrix}, \quad \mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}, \quad and \quad \mathbf{b} = \begin{bmatrix} 13 \\ -13 \\ -6 \end{bmatrix}.$$

In diesem Gleichungssystem ist die Anzahl der Gleichungen mit der Anzahl der Unbekannten identisch. Es wird als quadratisches Gleichungssystem bezeichnet. Für dieses System ist grundsätzlich eine eindeutige Lösung vorhanden. Das Gleichungssystem bildet die Schnittmenge der drei Ebenen des durch die drei Gleichungen dargestellten Koordinatensystems (x_1, x_2, x_3).

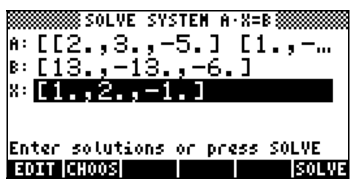
Zum Eingeben von Matrix **A** können Sie MatrixWriter aktivieren, während das Feld A: ausgewählt ist. In den folgenden Bildschirmabbildungen ist Matrix Writer für die Eingabe von Matrix **A** sowie die Eingabemaske für die numerische Lösung nach der Eingabe von Matrix **A** dargestellt (drücken Sie in Matrix Writer ENTER):



Drücken Sie ▼, um das Feld B: auszuwählen. Vektor b kann mit einfachen Klammern als Zeilenvektor eingegeben werden, d. h. [13,-13,-6] OK. Nachdem wir Matrix A und Vektor b eingegeben haben und das Feld X: markiert ist, können wir SOLVE drücken, um eine Lösung für dieses Gleichungssystem zu bestimmen:

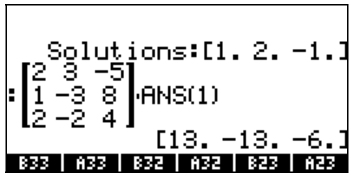


Die Lösung wird unten dargestellt.



Um die Lösung im Stack anzuzeigen, drücken Sie ENTER. Die Lösung lautet **x** = [1,2,-1].

Um die Lösung auf Richtigkeit zu überprüfen, geben Sie Matrix A ein und multiplizieren die Matrix mit diesem Lösungsvektor (Beispiel im algebraischen Modus):



Unterbestimmtes Gleichungssystem
Das lineare Gleichungssystem

$$2x_1 + 3x_2 - 5x_3 = -10,$$
$$x_1 - 3x_2 + 8x_3 = 85$$

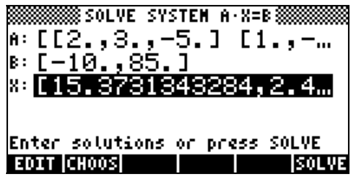
kann als Matrixgleichung $\mathbf{A}\cdot\mathbf{x} = \mathbf{b}$ beschrieben werden, wenn

$$\mathbf{A} = \begin{bmatrix} 2 & 3 & -5 \\ 1 & -3 & 8 \end{bmatrix}, \quad \mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}, \quad and \quad \mathbf{b} = \begin{bmatrix} -10 \\ 85 \end{bmatrix}.$$

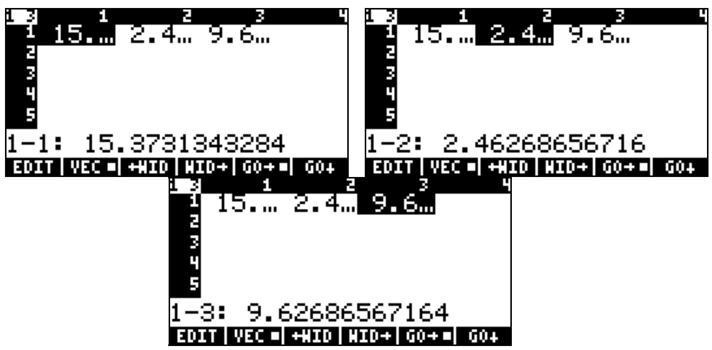
es über mehr Unbekannte als Gleichungen verfügt und daher nicht eindeutig bestimmt ist. Wir können die Bedeutung dieser Aussage veranschaulichen, wenn wir uns vorstellen, dass jede der linearen Gleichungen eine Ebene in einem dreidimensionalen kartesischen Koordinatensystem (x_1, x_2, x_3) darstellt. Die Lösung des obigen Gleichungssystems stellt die Schnittmenge zweier Ebenen im Raum dar. Wir wissen jedoch, dass die Schnittmenge zweier (nichtparalleler) Ebenen eine gerade Linie und nicht ein einzelner Punkt ist. Daher erfüllt mehr als ein Punkt die Bedingungen des Gleichungssystems. In diesem Sinn ist das Gleichungssystem nicht eindeutig bestimmt.

Wir suchen nun mit der numerischen Lösung nach einer Lösung dieses Gleichungssystems: ⇨ NUM.SLV ▼ ▼ ▼ OK. Geben Sie Matrix A und

Vektor b wie im vorherigen Beispiel veranschaulicht ein und drücken Sie SOLVE, wenn das Feld X: markiert ist:



Um ggf. Details des Lösungsvektors anzuzeigen, drücken Sie die Taste EDIT. Hierdurch wird MatrixWriter aktiviert. Verwenden Sie in MatrixWriter die Tasten mit dem Pfeil nach rechts bzw. nach links, um innerhalb des Vektors zu navigieren, z. B.



Die Lösung lautet somit **x** = [15.373, 2.4626, 9.6268].

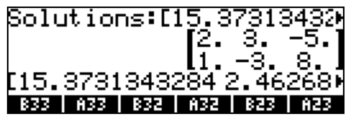
Um zur numerischen Lösung zurückzukehren, drücken Sie ENTER.

Das im Folgenden beschriebene Verfahren kann zum Kopieren von Matrix A und Lösungsvektor X in den Stack verwendet werden. Um die Richtigkeit der Lösung zu überprüfen, gehen Sie folgendermaßen vor:

- Drücken Sie ▲▲, um das Feld A: zu markieren.
- Drücken Sie NXT CALC ENTER, um Matrix A in den Stack zu kopieren.
- Drücken Sie OK, um zur numerischen Lösung zurückzukehren.

- Drücken Sie ▼ ▼ CALC ENTER, um den Lösungsvektor X in den Stack zu kopieren.
- Drücken Sie OK, um zur numerischen Lösung zurückzukehren.
- Drücken Sie ENTER, um zum Stack zurückzukehren.

Der Stack wird nun im ALG-Modus wie folgt angezeigt:

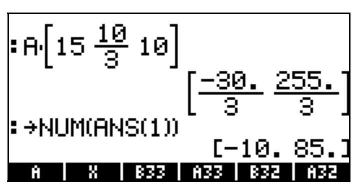


Speichern Sie nun das letzte Ergebnis in einer Variablen X und die Matrix in einer Variablen A:

Drücken Sie STO▶ ALPHA X ENTER, um den Lösungsvektor in der Variablen X zu speichern.
Drücken Sie ◀ ◀ ◀, um drei Ebenen des Stacks zu leeren.
Drücken Sie STO▶ ALPHA A ENTER, um die Matrix in der Variablen A zu speichern.

Überprüfen Sie nun die Lösung, indem Sie A × X ENTER drücken. Dies ergibt folgendes Ergebnis (drücken Sie ▼, um die Vektorelemente anzuzeigen): [-9.99999999999 85]. Dies unterscheidet sich nicht zu sehr vom ursprünglichen Vektor **b** = [-10 85].

Geben Sie außerdem A × [15,10/3,10] ENTER ↱ →NUM ENTER ein, also



Das Ergebnis bedeutet, dass **x** = [15,10/3,10] auch eine Lösung des Gleichungssystems darstellt, und bestätigt unsere Aussage, dass ein

Gleichungssystem, das über mehr Unbekannte als Gleichungen verfügt, nicht eindeutig bestimmt (unterbestimmt) ist.

Wie berechnet der Taschenrechner die zuvor dargestellte Lösung **x** = [15.37… 2.46… 9.62…]? Der Taschenrechner minimiert den Abstand von einem Punkt, der die Lösung darstellt, zu jeder der durch die Gleichungen im linearen Gleichungssystem dargestellten Ebenen. Der Taschenrechner verwendet die Methode der kleinste Quadrate , d. h., die Summe der Quadrate dieser Abstände bzw. Fehler wird minimiert.

Überbestimmtes Gleichungssystem

Das lineare Gleichungssystem

$$\begin{aligned} x_1 + 3x_2 &= 15,\\ 2x_1 - 5x_2 &= 5,\\ -x_1 + x_2 &= 22 \end{aligned}$$

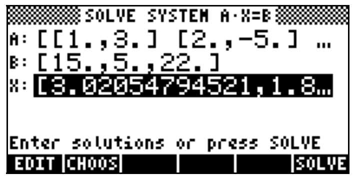
kann als Matrixgleichung **A**·**x** = **b** beschrieben werden, wenn

$$\mathbf{A} = \begin{bmatrix} 1 & 3 \\ 2 & -5 \\ -1 & 1 \end{bmatrix}, \quad \mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}, \quad and \quad \mathbf{b} = \begin{bmatrix} 15 \\ 5 \\ 22 \end{bmatrix}.$$

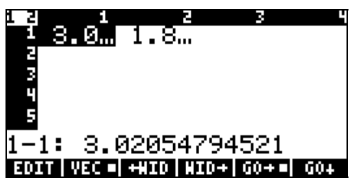
dieses System über mehr Gleichungen als Unbekannte verfügt (überbestimmtes Gleichungssystem). Für das System gibt es keine einzelne Lösung. Jede lineare Gleichung im oben dargestellten Gleichungssystem stellt eine gerade Linie in einem zweidimensionalen kartesischen Koordinatensystem (x_1, x_2) dar. Sofern zwei der drei Gleichungen des Systems nicht dieselbe Gleichung darstellen, besitzen die drei Linien mehr als einen Schnittpunkt. Daher ist die Lösung nicht eindeutig. Mithilfe einiger numerischer Algorithmen kann eine Lösung für das Gleichungssystem erzwungen werden, indem der Abstand vom mutmaßlichen Lösungspunkt zu jeder Linie des Gleichungssystems minimiert wird. Dies ist der von der numerischen Lösung des HP 49 G verwendete Ansatz.

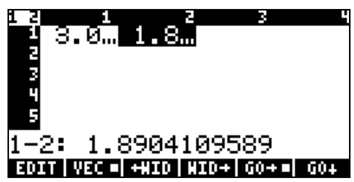
Wir suchen nun mit der numerischen Lösung nach einer Lösung dieses Gleichungssystems: ↱ NUM.SLV ▼ ▼ ▼ OK. Geben Sie Matrix A und

Vektor b wie im vorherigen Beispiel veranschaulicht ein und drücken Sie SOLVE, wenn das Feld X: markiert ist:



Um ggf. Details des Lösungsvektors anzuzeigen, drücken Sie die Taste EDIT. Hierdurch wird MatrixWriter aktiviert. Verwenden Sie in MatrixWriter die Tasten mit dem Pfeil nach rechts bzw. nach links, um innerhalb des Vektors zu navigieren, z. B.

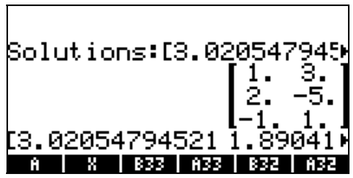




Drücken Sie ENTER, um zur numerischen Lösung zurückzukehren. Um die Richtigkeit der Lösung zu überprüfen, gehen Sie folgendermaßen vor:

- Drücken Sie ▲▲, um das Feld A: zu markieren.
- Drücken Sie NXT CALC ENTER, um Matrix A in den Stack zu kopieren.
- Drücken Sie OK, um zur numerischen Lösung zurückzukehren.
- Drücken Sie ▼ ▼ CALC ENTER, um den Lösungsvektor X in den Stack zu kopieren.
- Drücken Sie OK, um zur numerischen Lösung zurückzukehren.
- Drücken Sie ENTER, um zum Stack zurückzukehren.

Der Stack wird nun im ALG-Modus wie folgt angezeigt:



Speichern Sie nun das letzte Ergebnis in einer Variablen X und die Matrix in einer Variablen A:

Drücken Sie STO▶ ALPHA X ENTER, um den Lösungsvektor in der Variablen X zu speichern.
Drücken Sie ◄ ◄ ◄, um drei Ebenen des Stacks zu leeren.
Drücken Sie STO▶ ALPHA A ENTER, um die Matrix in der Variablen A zu speichern.

Überprüfen Sie nun die Lösung, indem Sie A × X ENTER drücken, so dass als Ergebnis der Vektor [8.6917… -3.4109… -1.1301…] angezeigt wird. Dies unterscheidet sich von [15 5 22], dem ursprünglichen Vektor **b**. Bei der „Lösung" handelt es sich einfach um den Punkt mit der geringsten Entfernung zu den drei durch die Gleichungen des Systems dargestellten Linien und nicht um eine exakte Lösung.

Lösung nach der Methode der kleinsten Quadrate (Funktion LSQ)

Mit der Funktion LSQ wird eine Lösung nach der Methode der kleinsten Quadrate eines linearen Gleichungssystems Ax = b nach den folgenden Kriterien ausgegeben:

- Wenn **A** eine quadratische nichtsinguläre Matrix ist (d. h., sie verfügt über eine inverse Matrix, oder ihre Determinante ist ungleich Null), gibt LSQ die exakte Lösung des linearen Gleichungssystems zurück.
- Wenn **A** keinen vollen Zeilenrang aufweist (unterbestimmtes Gleichungssystem), gibt LSQ aus einer unendlichen Anzahl von Lösungen die Lösung mit der minimalen euklidischen Länge zurück.
- Wenn **A** keinen vollen Spaltenrang aufweist (überbestimmtes Gleichungssystem), gibt LSQ die „Lösung" mit dem minimalen Residuum **e** = **A**·**x** – **b** zurück. Möglicherweise gibt es keine Lösung für das Gleichungssystem. Daher ist der zurückgegebene Wert keine

echte Lösung des Gleichungssystems, sondern lediglich der Wert mit dem kleinsten Residuum.

Die Eingangswerte für die Funktion LSQ sind Vektor **b** und Matrix **A** in dieser Reihenfolge. Die Funktion LSQ ist über den Funktionskatalog (CAT) verfügbar. Im Folgenden wiederholen wir die zuvor mit der numerischen Lösung ermittelten Lösungen mit der Funktion LSQ:

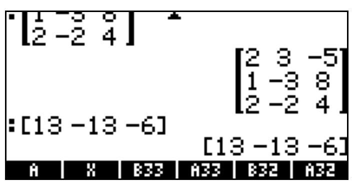
Quadratisches Gleichungssystem

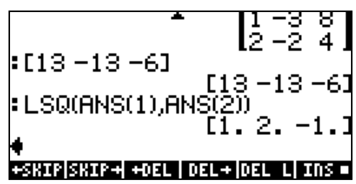
Gegeben sei das System

$$2x_1 + 3x_2 - 5x_3 = 13,$$
$$x_1 - 3x_2 + 8x_3 = -13,$$
$$2x_1 - 2x_2 + 4x_3 = -6$$

$$\mathbf{A} = \begin{bmatrix} 2 & 3 & -5 \\ 1 & -3 & 8 \\ 2 & -2 & 4 \end{bmatrix}, \quad \mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}, \quad and \quad \mathbf{b} = \begin{bmatrix} 13 \\ -13 \\ -6 \end{bmatrix}.$$

Die mit LSQ ermittelte Lösung wird unten dargestellt:





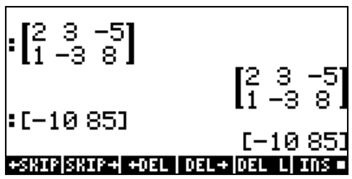
Unterbestimmtes Gleichungssystem

Gegeben sei das System

$$2x_1 + 3x_2 - 5x_3 = -10,$$
$$x_1 - 3x_2 + 8x_3 = 85$$

$$\mathbf{A} = \begin{bmatrix} 2 & 3 & -5 \\ 1 & -3 & 8 \end{bmatrix}, \quad \mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}, \quad and \quad \mathbf{b} = \begin{bmatrix} -10 \\ 85 \end{bmatrix}.$$

Die mit LSQ ermittelte Lösung wird unten dargestellt:



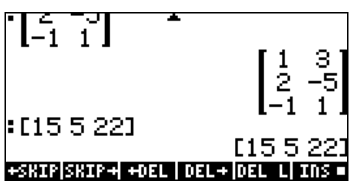
Überbestimmtes Gleichungssystem

Gegeben sei das System

$$x_1 + 3x_2 = 15,$$
$$2x_1 - 5x_2 = 5,$$
$$-x_1 + x_2 = 22$$

$$\mathbf{A} = \begin{bmatrix} 1 & 3 \\ 2 & -5 \\ -1 & 1 \end{bmatrix}, \quad \mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}, \quad and \quad \mathbf{b} = \begin{bmatrix} 15 \\ 5 \\ 22 \end{bmatrix}.$$

Die mit LSQ ermittelte Lösung wird unten dargestellt:



Vergleichen Sie diese drei Lösungen mit den Lösungen, die mit der numerischen Lösung berechnet wurden.

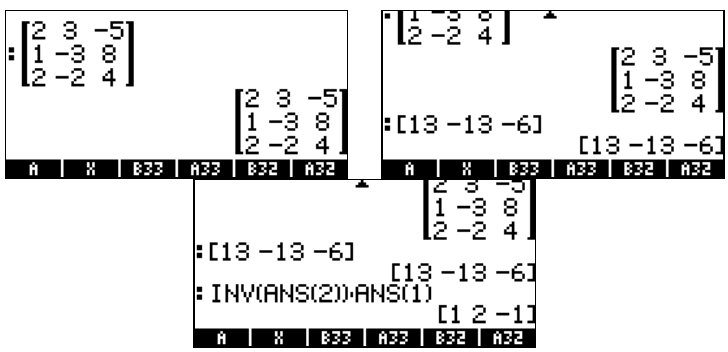
Lösung mit der inversen Matrix

Die Lösung des Gleichungssystems $\mathbf{A}\cdot\mathbf{x} = \mathbf{b}$, wobei $\mathbf{A}$ eine quadratische Matrix ist, lautet $\mathbf{x} = \mathbf{A}^{-1}\cdot\mathbf{b}$. Dieses Ergebnis entsteht durch Multiplikation der ersten Gleichung mit $\mathbf{A}^{-1}$, also $\mathbf{A}^{-1}\cdot\mathbf{A}\cdot\mathbf{x} = \mathbf{A}^{-1}\cdot\mathbf{b}$. Definitionsgemäß ist $\mathbf{A}^{-1}\cdot\mathbf{A} = \mathbf{I}$, daher schreiben wir $\mathbf{I}\cdot\mathbf{x} = \mathbf{A}^{-1}\cdot\mathbf{b}$. Darüber hinaus ist $\mathbf{I}\cdot\mathbf{x} = \mathbf{x}$, somit gilt $\mathbf{x} = \mathbf{A}^{-1}\cdot\mathbf{b}$.

Für das zuvor verwendete Beispiel

$$
\begin{aligned}
2x_1 + 3x_2 - 5x_3 &= 13\\
x_1 - 3x_2 + 8x_3 &= -13\\
2x_1 - 2x_2 + 4x_3 &= -6
\end{aligned}
$$

ermitteln wir mit dem Taschenrechner die Lösung wie folgt:



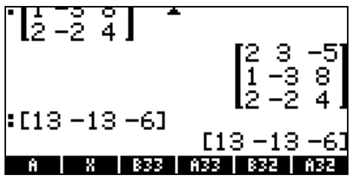
Dies ist mit dem zuvor ermittelten Ergebnis identisch.

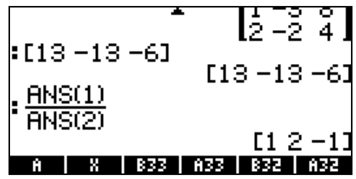
Lösung durch „Division" von Matrizen

Obwohl die Division für Matrizen nicht definiert ist, können wir mithilfe der Taste ÷ des Taschenrechners Vektor $\mathbf{b}$ durch Matrix $\mathbf{A}$ „dividieren", um in der Matrixgleichung $\mathbf{A}\cdot\mathbf{x} = \mathbf{b}$ eine Lösung für $\mathbf{x}$ zu finden. Es handelt sich hier um eine willkürliche Erweiterung der algebraischen Division von Matrizen, d. h., aufgrund von $\mathbf{A}\cdot\mathbf{x} = \mathbf{b}$ wagen wir zu schreiben $\mathbf{x} = \mathbf{b}/\mathbf{A}$ (Mathematiker würden schaudern!). Dies wird selbstverständlich als $(1/\mathbf{A})\cdot\mathbf{b} = \mathbf{A}^{-1}\cdot\mathbf{b}$ interpretiert und entspricht der Verwendung der Inversen von $\mathbf{A}$ im vorherigen Abschnitt. Das Verfahren für die „Division" von **b** durch **A** wird unten für

$$
\begin{aligned}
2x_1 + 3x_2 - 5x_3 &= 13,\\
x_1 - 3x_2 + 8x_3 &= -13,\\
2x_1 - 2x_2 + 4x_3 &= -6
\end{aligned}
$$

veranschaulicht. In den folgenden Bildschirmabbildungen wird das Verfahren dargestellt:





Es handelt sich um dieselbe Lösung, die oben mit der inversen Matrix ermittelt wurde.

Lösen mehrerer Gruppen von Gleichungen mit derselben Koeffizientenmatrix

Angenommen, Sie möchten die folgenden drei Gruppen von Gleichungen lösen:

X + 2Y + 3Z = 14,	2X + 4Y + 6Z = 9,	2X + 4Y + 6Z = -2,
3X - 2Y + Z = 2,	3X - 2Y + Z = -5,	3X - 2Y + Z = 2,
4X + 2Y - Z = 5,	4X + 2Y - Z = 19,	4X + 2Y -Z = 12.

Die drei Gleichungssysteme können als eine einzige Matrixgleichung dargestellt werden: **A**·**X** = **B**, mit

$$
\mathbf{A} = \begin{bmatrix} 1 & 2 & 3 \\ 3 & -2 & 1 \\ 4 & 2 & -1 \end{bmatrix}, \quad \mathbf{X} = \begin{bmatrix} X_{(1)} & X_{(2)} & X_{(3)} \\ Y_{(1)} & Y_{(2)} & Y_{(3)} \\ Z_{(1)} & Z_{(2)} & Z_{(3)} \end{bmatrix},
$$

$$\mathbf{B} = \begin{bmatrix} 14 & 9 & -2 \\ 2 & -5 & 2 \\ 5 & 19 & 12 \end{bmatrix}.$$

Die Indizes in den Variablennamen X, Y und Z geben an, auf welches Gleichungssystem sie sich beziehen. Zur Lösung dieses erweiterten Systems verwenden wir im RPN-Modus folgendes Verfahren:

```
[[14,9,-2],[2,-5,2],[5,19,12]] ENTER
[[1,2,3],[3,-2,1],[4,2,-1]] ENTER ÷
```

Das Ergebnis dieser Operation lautet:

$$\mathbf{X} = \begin{bmatrix} 1 & 2 & 2 \\ 2 & 5 & 1 \\ 3 & -1 & -2 \end{bmatrix}.$$

Gauß- und Gauß-Jordan-Elimination

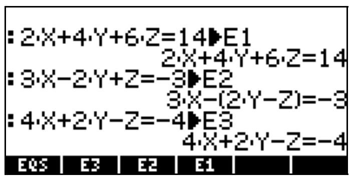
Bei der Gauß-Elimination wird eine quadratische Koeffizientenmatrix, die einem System mit n linearen Gleichungen und n Unbekannten angehört, über mehrere Zeilenoperationen zu einer oberen Dreiecksmatrix (*Staffelform*) reduziert. Dieses Verfahren wird als *Vorwärtssubstitution* bezeichnet. Aufgrund der Reduzierung der Koeffizientenmatrix zu einer oberen Dreiecksmatrix kann mit einem als *Rückwärtssubstitution* bezeichneten Verfahren, bei dem jeweils nur eine Gleichung bearbeitet wird, eine Lösung für alle *n* Unbekannten ermittelt werden.

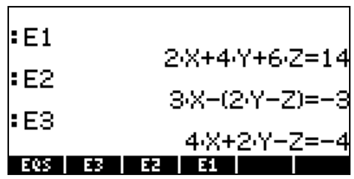
Beispiel für die Gauß-Elimination mit Gleichungen

Zur Veranschaulichung der Gauß-Elimination verwenden wir folgendes System mit 3 Gleichungen und 3 Unbekannten:

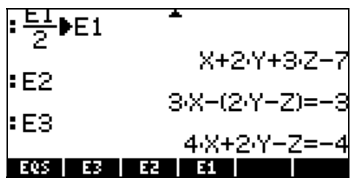
$$2X + 4Y + 6Z = 14,$$
$$3X - 2Y + Z = -3,$$
$$4X + 2Y - Z = -4.$$

Wir speichern diese Gleichungen mit dem Taschenrechner in den Variablen E1, E2 bzw. E3, wie unten dargestellt. Für Backup-Zwecke wurde außerdem eine Liste mit den drei Gleichungen erstellt und in der Variablen EQS gespeichert. Falls eine fehlerhafte Eingabe erfolgt, sind die Gleichungen somit dennoch für den Benutzer verfügbar.

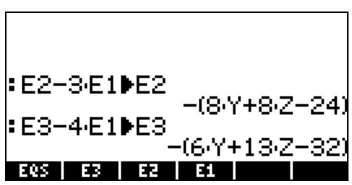


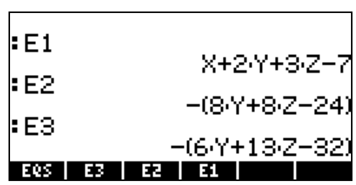


Zu Beginn der Vorwärtssubstitution dividieren wir die erste Gleichung (E1) durch 2, speichern sie in E1 und zeigen die drei Gleichungen erneut an:

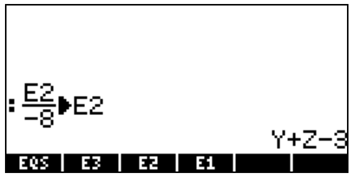


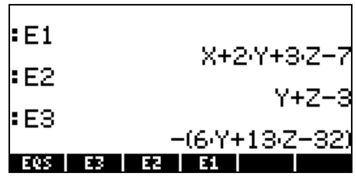
Anschließend ersetzen wir die zweite Gleichung E2 durch (Gleichung 2-3×Gleichung 1, also E1-3×E2) und die dritte Gleichung durch (Gleichung 3-4×Gleichung 1) und erhalten



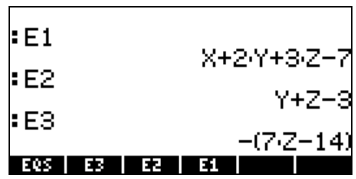


Dann dividieren wir die zweite Gleichung durch -8 und erhalten





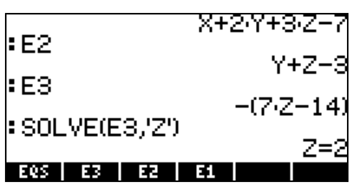
Anschließend ersetzen wir die dritte Gleichung E3 durch (Gleichung 3+6×Gleichung 2, also E2+6×E3) und erhalten

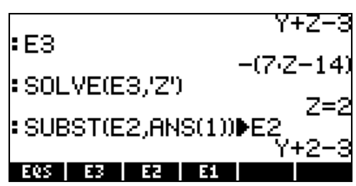


Beachten Sie, dass der Taschenrechner beim Ausführen einer Linearkombination von Gleichungen das Ergebnis in einen Ausdruck auf der linken Seite der Gleichung ändert, d. h. Ausdruck = 0. Die letzte Gruppe von Gleichungen wird somit als folgende äquivalente Gruppe von Gleichungen interpretiert:

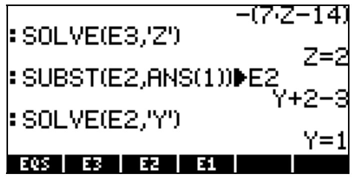
$$X + 2Y + 3Z = 7,$$
$$Y + Z = 3,$$
$$-7Z = -14.$$

Bei der Gauß-Elimination besteht die Rückwärtssubstitution darin, dass die Werte der Unbekannten ermittelt werden, wobei mit der letzten Gleichung begonnen und der Vorgang mit den jeweils oberen Gleichungen fortgesetzt wird. Wir ermitteln daher zuerst Z:

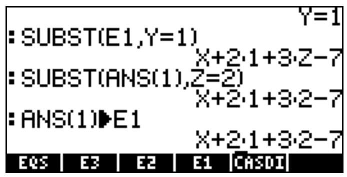


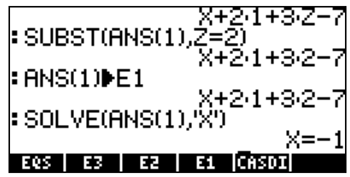


Dann setzen wir in Gleichung 2 (E2) für Z=2 ein und ermitteln Y in E2:



Anschließend setzen wir in E1 für Z=2 und für Y=1 ein und ermitteln X in E1:





Die Lösung lautet somit X = -1, Y = 1, Z = 2.

Beispiel für die Gauß-Elimination mit Matrizen

Das im obigen Beispiel verwendete Gleichungssystem kann als Matrixgleichung **A**·**x** = **b** dargestellt werden, wenn wir schreiben:

$$\mathbf{A} = \begin{pmatrix} 2 & 4 & 6 \\ 3 & -2 & 1 \\ 4 & 2 & -1 \end{pmatrix}, \quad \mathbf{x} = \begin{bmatrix} X \\ Y \\ Z \end{bmatrix}, \quad \mathbf{b} = \begin{bmatrix} 14 \\ -3 \\ -4 \end{bmatrix}.$$

Um mithilfe der Gauß-Elimination eine Lösung für die Matrix des Gleichungssystems zu erhalten, erstellen wir zunächst eine **A** entsprechende, so genannte <u>*erweiterte Matrix*</u>, also

$$\mathbf{A}_{aug} = \left(\begin{array}{ccc|c} 2 & 4 & 6 & 14 \\ 3 & -2 & 1 & -3 \\ 4 & 2 & -1 & -4 \end{array}\right)$$

Bei Matrix $\mathbf{A}_{aug}$ handelt es sich um die ursprüngliche Matrix **A** mit einer neuen Zeile, die den Elementen von Vektor **b** entspricht und rechts von der äußersten rechten Spalte von **A** eingefügt (d. h. erweitert) wird.

Nachdem die erweiterte Matrix gebildet wurde, können wir mit ihr Zeilenoperationen durchführen, mit denen die ursprüngliche Matrix A zu einer oberen Dreiecksmatrix reduziert wird. Hierfür verwenden wir den RPN-Modus (MODE +/- OK), wobei der Systemflag 117 auf das Menü SOFT gesetzt ist. Verwenden Sie für den Taschenrechner folgende Tastenkombinationen. Geben Sie zunächst die erweiterte Matrix ein und erstellen Sie im Stack eine Kopie der Matrix (Dieser Schritt ist nicht erforderlich, dient jedoch der Absicherung, damit Sie über eine Kopie der erweiterten Matrix verfügen, falls Ihnen bei der durchzuführenden Vorwärtssubstitution ein Fehler unterläuft.):

`[[2,4,6,14],[3,-2,1,-3],[4,2,-1,-4]]` ENTER ENTER

Speichern Sie die erweiterte Matrix in der Variablen AAUG: ' ALPHA ALPHA A A U G ALPHA STO▶

Wenn eine Kopie der erweiterten Matrix im Stack vorhanden ist, drücken Sie ← MTH MATRX ROW, um das Menü ROW zu aktivieren. Wenden Sie anschließend die folgenden Zeilenoperationen auf die erweiterte Matrix an:
Multiplizieren Sie Zeile 1 mit 1/2: 2 1/x 1 RCI

Multiplizieren Sie Zeile 1 mit -3 und fügen Sie sie Zeile 2 hinzu, indem Sie diese ersetzen: 3 +/- SPC 1 SPC 2 RCIJ

Multiplizieren Sie Zeile 1 mit -4 und fügen Sie sie Zeile 3 hinzu, indem Sie diese ersetzen: 4 +/- SPC 1 SPC 3 RCIJ

Multiplizieren Sie Zeile 2 mit -1/8: 8 +/- 1/x 2 RCI
Multiplizieren Sie Zeile 2 mit 6 und fügen Sie sie Zeile 3 hinzu, indem Sie diese ersetzen: 6 SPC 2 SPC 3 RCIJ

Wenn Sie diese Operationen manuell durchführen, gehen Sie folgendermaßen vor:

$$\mathbf{A}_{aug} = \left(\begin{array}{ccc|c} 2 & 4 & 6 & 14 \\ 3 & -2 & 1 & -3 \\ 4 & 2 & -1 & -4 \end{array}\right) \cong \left(\begin{array}{ccc|c} 1 & 2 & 3 & 7 \\ 3 & -2 & 1 & -3 \\ 4 & 2 & -1 & -4 \end{array}\right)$$

$$\mathbf{A}_{aug} \cong \left(\begin{array}{ccc|c} 1 & 2 & 3 & 7 \\ 0 & -8 & -8 & -24 \\ 0 & -6 & -13 & -32 \end{array}\right) \cong \left(\begin{array}{ccc|c} 1 & 2 & 3 & 7 \\ 0 & 1 & 1 & 3 \\ 0 & -6 & -13 & -32 \end{array}\right)$$

$$\mathbf{A}_{aug} \cong \left(\begin{array}{ccc|c} 1 & 2 & 3 & 7 \\ 0 & 1 & 1 & 3 \\ 0 & 0 & -7 & -14 \end{array}\right)$$

Das Symbol ≅ („entspricht“) gibt an, dass der folgende Ausdruck mit der vorherigen Matrix, auf die einige Zeilenoperationen (bzw. Spaltenoperationen) angewendet wurden, äquivalent ist.

Die resultierende Matrix ist eine obere Dreiecksmatrix und äquivalent mit der Gruppe von Gleichungen

X +2Y+3Z = 7,
Y+ Z = 3,
-7Z = -14

Diese können nun wie im vorherigen Beispiel durch Rückwärtssubstitution einzeln nacheinander gelöst werden.

Gauß-Jordan-Elimination mit Matrizen

Bei der Gauß-Jordan-Elimination werden die Zeilenoperationen in der aus der Rückwärtssubstitution resultierenden oberen Dreiecksmatrix fortgesetzt, bis anstelle der ursprünglichen Matrix **A** eine Einheitsmatrix gebildet wurde. Beispielsweise können wir im gerade dargestellten Fall die Zeilenoperationen wie folgt fortsetzen:

Multiplizieren Sie Zeile 3 mit -1/7: [7] [+/-] [1/x] [3] [RCI]

Multiplizieren Sie Zeile 3 mit -1 und fügen Sie sie Zeile 2 hinzu, indem Sie diese ersetzen: 1 +/- SPC 3 SPC 2 RCIJ

Multiplizieren Sie Zeile 3 mit -3 und fügen Sie sie Zeile 1 hinzu, indem Sie diese ersetzen: 3 +/- SPC 3 SPC 1 RCIJ

Multiplizieren Sie Zeile 2 mit -2 und fügen Sie sie Zeile 1 hinzu, indem Sie diese ersetzen: 2 +/- SPC 2 SPC 1 RCIJ

Wenn Sie diesen Vorgang manuell durchführen, ergeben sich folgende Schritte:

$$\mathbf{A}_{aug} = \left(\begin{array}{ccc|c} 1 & 2 & 3 & 7 \\ 0 & 1 & 1 & 3 \\ 0 & 0 & -7 & -14 \end{array}\right) \cong \left(\begin{array}{ccc|c} 1 & 2 & 3 & 7 \\ 0 & 1 & 1 & 3 \\ 0 & 0 & 1 & 2 \end{array}\right) \cong \left(\begin{array}{ccc|c} 1 & 2 & 3 & 7 \\ 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 2 \end{array}\right)$$

$$A_{aug} \cong \left(\begin{array}{ccc|c} 1 & 2 & 0 & 1 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 2 \end{array}\right) \cong \left(\begin{array}{ccc|c} 1 & 0 & 0 & -1 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 2 \end{array}\right).$$

Pivotisierung

Wenn Sie die Zeilenoperationen in den oben dargestellten Beispielen sorgfältig untersuchen, werden Sie feststellen, dass durch viele dieser Operationen eine Zeile durch das entsprechende Element in der Hauptdiagonalen dividiert wird. Dieses Element wird als Pivot-Element bezeichnet. In zahlreichen Fällen kann das Pivot-Element den Wert Null annehmen, so dass die Zeile nicht durch ihr Pivot-Element dividiert werden kann. Zur Vereinfachung der numerischen Lösung eines Gleichungssystems mit der Gauß- oder Gauß-Jordan-Elimination empfiehlt es sich außerdem, als Pivot-Element das Element mit dem größten absoluten Wert in einer Spalte zu verwenden. In diesen Fällen vertauschen wir Zeilen vor der Anwendung der Zeilenoperationen. Dieses Vertauschen von Zeilen wird als Teilpivotisierung bezeichnet. Zum Befolgen dieser Empfehlung müssen häufig Zeilen in der erweiterten Matrix vertauscht werden, wenn eine Gauß- oder Gauß-Jordan-Elimination durchgeführt wird.

Bei der Pivotisierung während einer Matrixelimination können Sie die numerische Lösung noch weiter vereinfachen, indem Sie das Element mit dem größten absoluten Wert in der betreffenden Spalte bzw. Zeile als Pivot-Element auswählen. Dies erfordert möglicherweise, dass bei einigen Pivotisierungsoperationen nicht nur Zeilen, sondern auch Spalten vertauscht werden. Wenn bei der Pivotisierung Zeilen- und Spaltentausch zulässig ist, wird dieses Verfahren als Totalpivotisierung bezeichnet.

Beim Vertauschen von Zeilen und Spalten bei der Teil- bzw. Totalpivotisierung ist es erforderlich, die Tauschvorgänge aufzuzeichnen, da hierbei die Anordnung der Unbekannten in der Lösung geändert wird. Eine Möglichkeit zum Aufzeichnen der Spaltentauschvorgänge bei der Teil- bzw. Totalpivotisierung besteht darin, zu Beginn des Verfahrens eine *Permutationsmatrix* $\mathbf{P} = \mathbf{I}_{n\times n}$ zu erstellen. Jeder in der erweiterten Matrix $\mathbf{A}_{aug}$ erforderliche Zeilen- oder Spaltentausch wird auch in der Permutationsmatrix als Zeilen- bzw. Spaltentausch registriert. Wenn die Lösung ermittelt wurde, multiplizieren wir die Permutationsmatrix mit dem Vektor der Unbekannten $\mathbf{x}$, um die richtige Anordnung der Unbekannten in der Lösung zu erhalten. Mit anderen Worten, die endgültige Lösung ist durch $\mathbf{P}\cdot\mathbf{x} = \mathbf{b}'$ definiert, wobei $\mathbf{b}'$ die letzte Spalte in der erweiterten Matrix darstellt, nachdem die Lösung ermittelt wurde.

Beispiel für Gauß-Jordan-Elimination mit Totalpivotisierung

Die Totalpivotisierung wird anhand eines Beispiels veranschaulicht. Lösen Sie das folgende Gleichungssystem unter Verwendung von Totalpivotisierung und Gauß-Jordan-Elimination:

$$X + 2Y + 3Z = 2,$$
$$2X + 3Z = -1,$$
$$8X + 16Y - Z = 41.$$

Die erweiterte Matrix und die Permutationsmatrix lauten wie folgt:

$$\mathbf{A}_{aug} = \begin{bmatrix} 1 & 2 & 3 & 2 \\ 2 & 0 & 3 & -1 \\ 8 & 16 & -1 & 41 \end{bmatrix}, \quad \mathbf{P} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}.$$

Speichern Sie die erweiterte Matrix in der Variablen AAUG und drücken Sie dann ‚ AAUG, um die erweiterte Matrix in den Stack zu kopieren. Wir möchten, dass der Befehl CSWP (Spalten vertauschen) verfügbar bleibt, für den wir Folgendes eingeben: ‚N ALPHA ALPHA C S ALPHA (CSWP suchen), OK. Sie erhalten eine Fehlermeldung. Drücken Sie $ und ignorieren Sie die Meldung.
Machen Sie anschließend das Menü ROW verfügbar, indem Sie
„MATRICES CREATE ROW drücken.

Nun können wir mit der Gauß-Jordan-Elimination mit Totalpivotisierung beginnen. Wir müssen die Permutationsmatrix per Hand aufzeichnen, tragen Sie also die oben dargestellte Matrix **P** in Ihr Notizbuch ein.

Zunächst überprüfen wir das Pivot-Element a_{11}. Wir stellen fest, dass das Element mit dem größten absoluten Wert in der ersten Zeile und ersten Spalte der Wert a_{31} = 8 ist. Da diese Zahl als Pivot-Element verwendet werden soll, vertauschen wir die Zeilen 1 und 3 mit dem Befehl 1 SPC 3 NXT RSWP. Die erweiterte Matrix und die Permutationsmatrix lauten nun:

$$\begin{bmatrix} 8 & 16 & -1 & 41 \\ 2 & 0 & 3 & -1 \\ 1 & 2 & 3 & 2 \end{bmatrix} \qquad \begin{bmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Beim Überprüfen des Pivot-Elements an Position (1,1) stellen wir nun fest, dass 16 besser als Pivot-Element geeignet ist als 8. Daher führen wir mit folgendem Befehl einen Spaltentausch durch: 1 SPC 2 ‚N OK.
RSWP. Die erweiterte Matrix und die Permutationsmatrix lauten nun:

$$\begin{bmatrix} 16 & 8 & -1 & 41 \\ 0 & 2 & 3 & -1 \\ 2 & 1 & 3 & 2 \end{bmatrix} \qquad \begin{bmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}$$

Der größte mögliche Wert befindet sich jetzt an Position (1,1), d. h., wir haben an Position (1,1) eine Totalpivotisierung durchgeführt. Anschließend dividieren wir durch das Pivot-Element:
[1] [6] [1/x] [1] [NXT] [RCI] . Die Permutationsmatrix bleibt unverändert, doch die erweiterte Matrix lautet nun:

$$\begin{bmatrix} 1 & 1/2 & -1/16 & 41/16 \\ 0 & 2 & 3 & -1 \\ 2 & 1 & 3 & 2 \end{bmatrix} \qquad \begin{bmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}$$

Als Nächstes entfernen wir die 2 aus Position (3,2):
[2] [+/-] [SPC] [1] [SPC] [3] [RCIJ]

$$\begin{bmatrix} 1 & 1/2 & -1/16 & 41/16 \\ 0 & 2 & 3 & -1 \\ 0 & 0 & 25/8 & -25/8 \end{bmatrix} \qquad \begin{bmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}$$

Nachdem wir die Elemente von Spalte 1 unter dem Pivot-Element mit Nullen aufgefüllt haben, überprüfen wir das Pivot-Element an Position (2,2). Wir stellen fest, dass die Zahl 3 an Position (2,3) besser als Pivot-Element geeignet ist und vertauschen daher Spalte 2 und 3 durch die Eingabe: [2] [SPC] [3] [→] CAT [OK]

$$\begin{bmatrix} 1 & -1/16 & 1/2 & 41/16 \\ 0 & 3 & 2 & -1 \\ 0 & 25/8 & 0 & -25/82 \end{bmatrix} \qquad \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Bei Überprüfung des Pivot-Elements an Position (2,2) stellen wir fest, dass der Wert 25/8 an Position (3,2) größer als 3 ist. Wir vertauschen daher Spalte 2 und 3 durch die Eingabe: [2] [SPC] [3] [NXT] [RSWP]

$$\begin{bmatrix} 1 & -1/16 & 1/2 & 41/16 \\ 0 & 25/8 & 0 & -25/8 \\ 0 & 3 & 2 & -1 \end{bmatrix} \qquad \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix}$$

Nun können wir Spalte 2 durch das Pivot-Element 25/8 dividieren, indem wir ['] [8] [÷] [2] [5] [▶] [SPC] [2] [NXT] [RCI] eingeben.

$$\begin{bmatrix} 1 & -1/16 & 1/2 & 41/16 \\ 0 & 1 & 0 & -1 \\ 0 & 3 & 2 & -1 \end{bmatrix} \quad \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix}$$

Anschließend entfernen wir die 3 aus Position (3,2) durch folgende Eingabe: [3] [+/-] [SPC] [2] [SPC] [3] [RCIJ]

$$\begin{bmatrix} 1 & -1/16 & 1/2 & 41/16 \\ 0 & 1 & 0 & -1 \\ 0 & 0 & 2 & 2 \end{bmatrix} \quad \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix}$$

Nachdem wir die Stellen unter dem Pivot-Element mit Nullen aufgefüllt haben, überprüfen wir das Pivot-Element an Position (3,3). Der aktuelle Wert 2 ist größer als 1/2 oder 0, daher lassen wir ihn unverändert. Wir dividieren die ganze dritte Reihe durch 2, um das Pivot-Element in 1 umzuwandeln, indem wir folgende Eingabe verwenden: [2] [1/x] [3] [RCI]

$$\begin{bmatrix} 1 & -1/16 & 1/2 & 41/16 \\ 0 & 1 & 0 & -1 \\ 0 & 0 & 1 & 1 \end{bmatrix} \quad \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix}$$

Anschließend entfernen wir 1/2 an Position (1,3) durch die Eingabe: [2] [1/x] [+/-] [SPC] [3] [SPC] [1] [RCIJ]

$$\begin{bmatrix} 1 & -1/16 & 0 & 33/16 \\ 0 & 1 & 0 & -1 \\ 0 & 0 & 1 & 1 \end{bmatrix} \quad \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix}$$

Schließlich entfernen wir -1/16 aus Position (1,2) durch folgende Eingabe: [1] [6] [1/x] [SPC] [2] [SPC] [1] [RCIJ]

$$\begin{bmatrix} 1 & 0 & 0 & 2 \\ 0 & 1 & 0 & -1 \end{bmatrix} \quad \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

0 0 1 1 1 0 0

Nun verfügen wir über eine Einheitsmatrix in dem der ursprünglichen Koeffizientenmatrix A entsprechenden Abschnitt der erweiterten Matrix und können mithilfe des in Permutationsmatrix **P** codierten Zeilen- und Spaltentausches die Lösung ermitteln. Wir bestimmen den Vektor der Unbekannten **x**, den Vektor der geänderten Unabhängigen **b**' und die Permutationsmatrix **P** wie folgt:

$$\mathbf{x} = \begin{bmatrix} X \\ Y \\ Z \end{bmatrix}, \quad \mathbf{b'} = \begin{bmatrix} 2 \\ -1 \\ 1 \end{bmatrix}, \quad \mathbf{P} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix}.$$

Die Lösung lautet **P**·**x**=**b**′ oder

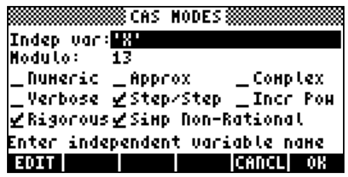
$$\begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} 3 \\ -1 \\ 1 \end{bmatrix}.$$

Dies ergibt

$$\begin{bmatrix} Y \\ Z \\ X \end{bmatrix} = \begin{bmatrix} 3 \\ -1 \\ 1 \end{bmatrix}.$$

Schrittweises Verfahren für den Taschenrechner zum Lösen linearer Gleichungssysteme

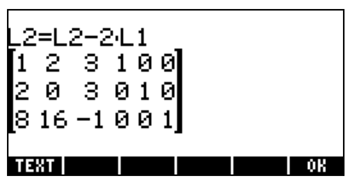
Bei dem gerade erläuterten Beispiel handelt es sich natürlich um ein schrittweises, vom Benutzer durchgeführtes Verfahren zur Lösung linearer Gleichungssysteme mit Totalpivotisierung für die Gauß-Jordan-Elimination. Sie können das vom Taschenrechner verwendete schrittweise Verfahren zum Lösen eines Gleichungssystems ohne Benutzereingaben anzeigen, indem Sie in einem CAS MODES-Fenster des Taschenrechners die Option Step/Step wie folgt auswählen.



Verwenden Sie dann für dieses Beispiel im RPN-Modus folgende Eingabe:

`[2,-1,41]` ENTER `[[1,2,3],[2,0,3],[8,16,-1]]` ENTER ÷

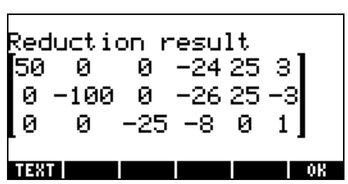
Der Taschenrechner zeigt eine erweiterte Matrix an, die aus der Koeffizientenmatrix **A** und der Einheitsmatrix **I** besteht, während gleichzeitig die nächste Berechnung angezeigt wird.



L2 = L2-2·L1 bedeutet „Reihe 2 (L2) durch die Operation L2-2 L1 ersetzen“. Bei der manuellen Ausführung dieser Operation würde dies folgender Eingabe entsprechen: 2 +/- SPC 1 SPC 1 RCI. Drücken Sie OK und beachten Sie die auf dem Bildschirm des Taschenrechners angezeigten Operationen. Es wird die Ausführung der folgenden Operationen angezeigt:

L3=L3-8·L1, L1 = 2·L1–1·L2, L1=25·L1–3·L3, L2 = 25·L2-3·L3

und schließlich die Meldung, dass das Ergebnis der Reduzierung („Reduction result“) angezeigt wird.



Wenn Sie OK drücken, gibt der Taschenrechner das Endergebnis [1 2 –1] aus.

Schrittweises Berechnen der inversen Matrix

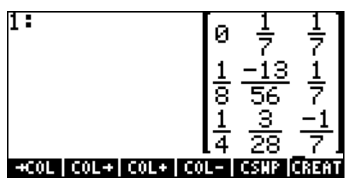
Die Berechnung einer inversen Matrix kann als Berechnung der Lösung eines erweiterten Systems [**A** | **I**] betrachtet werden. Beispielsweise würden wir für Matrix **A** aus dem vorherigen Beispiel die erweiterte Matrix wie folgt schreiben:

$$\mathbf{A}_{aug(I)} = \left[\begin{array}{ccc|ccc} 1 & 2 & 3 & 1 & 0 & 0 \\ 3 & -2 & 1 & 0 & 1 & 0 \\ 4 & 2 & -1 & 0 & 0 & 1 \end{array}\right].$$

Um die Zwischenschritte bei der Berechnung sowie die Inverse anzuzeigen, geben Sie einfach Matrix **A** aus dem vorherigen Beispiel ein und drücken Sie 1/x, während im CAS-MODES-Fenster des Taschenrechners die Option Step/Step aktiviert ist. Verwenden Sie folgende Eingabe:

`[[ 1,2,3],[3,-2,1],[4,2,-1]]` ENTER 1/x

Nach Ausführung der einzelnen Schritte lautet die ausgegebene Lösung:



Der Taschenrechner hat nicht eigentlich eine Gauß-Jordan-Elimination mit Totalpivotisierung angezeigt, sondern ein Verfahren zum Berechnen der Inversen einer Matrix mithilfe einer Gauß-Jordan-Elimination ohne Pivotisierung. Dieses Verfahren zum Berechnen der Inversen beruht auf der erweiterten Matrix $(\mathbf{A}_{aug})_{n\times n}$ = [$\mathbf{A}_{n\times n}$ |$\mathbf{I}_{n\times n}$].

Der Taschenrechner zeigt die Schritte bis zu dem Punkt an, an dem die linke Seite der erweiterten Matrix in eine Diagonalmatrix umgewandelt wurde. Nun besteht der letzte Schritt im Dividieren jeder Zeile durch das entsprechende Pivot-Element der Hauptdiagonalen. Mit anderen Worten, der Taschenrechner hat $(\mathbf{A}_{aug})_{n\times n}$ = [$\mathbf{A}_{n\times n}$ |$\mathbf{I}_{n\times n}$] in [$\mathbf{I}$ |$\mathbf{A}^{-1}$] konvertiert.

Inverse Matrizen und Determinanten

Beachten Sie, dass alle Elemente in der oben berechneten inversen Matrix durch den Wert 56 oder einen seiner Faktoren (28, 7, 8, 4 oder 1) dividiert werden. Wenn Sie die Determinante von Matrix **A** berechnen, erhalten Sie $det(\mathbf{A}) = 56$.

Wir können $\mathbf{A}^{-1} = \mathbf{C}/det(\mathbf{A})$ schreiben, wobei **C** folgende Matrix darstellt:

$$\mathbf{C} = \begin{bmatrix} 0 & 8 & 8 \\ 7 & -13 & 8 \\ 14 & 6 & -8 \end{bmatrix}.$$

Bei dem Ergebnis $(\mathbf{A}^{-1})_{n\times n} = \mathbf{C}_{n\times n} / det(\mathbf{A}_{n\times n})$ handelt es sich um ein generisches Ergebnis, das auf eine beliebige nichtsinguläre Matrix **A** zutrifft. Auf der Grundlage des Gauß-Jordan-Algorithmus kann eine allgemeine Darstellung der Elemente von **C** geschrieben werden.

Aufgrund der oben skizzierten Gleichung $\mathbf{A}^{-1} = \mathbf{C}/\det(\mathbf{A})$ ist die inverse Matrix $\mathbf{A}^{-1}$ nicht definiert, wenn $det(\mathbf{A}) = 0$. Die Bedingung $det(\mathbf{A}) = 0$ definiert somit auch eine singuläre Matrix.

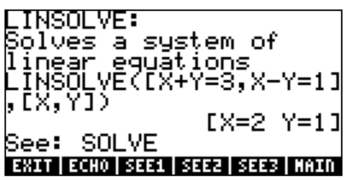
Lösung linearer Gleichungssysteme mit den Taschenrechner-funktionen

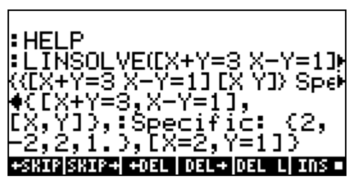
Die einfachste Möglichkeit zum Lösen eines linearen Gleichungssystems $\mathbf{A}\cdot\mathbf{x} = \mathbf{b}$ mit dem Taschenrechner besteht darin, **b** einzugeben, **A** einzugeben und anschließend die Divisionsfunktion / zu verwenden. Wenn das lineare Gleichungssystem überbestimmt oder unterbestimmt ist, kann mit der Funktion LSQ (Least SQares, kleinste Quadrate) eine „Lösung" ermittelt werden. Mit den Funktionen des Menüs MATRICES' LINEAR SYSTEMS, das über ← MATRICES aufgerufen werden kann (setzen Sie Systemflag 117 auf die Felder CHOOSE), bietet der Taschenrechner jedoch andere Möglichkeiten zum Lösen linearer Gleichungssysteme.

Die Funktionen dieses Menüs lauten LINSOLVE, REF, rref, RREF und SYST2MAT.

Funktion LINSOLVE

Als Argumente der Funktion LINSOLVE werden ein Feld von Gleichungen und ein Vektor verwendet, der die Namen der Unbekannten enthält. Die Funktion ermittelt die Lösung linearer Gleichungssysteme. In den folgenden Fenstern wird der Eintrag der Hilfefunktion (siehe Kapitel 1) für LINSOLVE und das zugehörige, im Eintrag aufgeführte Beispiel dargestellt. Im rechten Fenster wird das unter Verwendung des Befehlszeileneditors (drücken Sie zum Aktivieren) erzeugte Ergebnis angezeigt:





Es folgt ein weiteres Beispiel im ALG-Modus. Geben Sie Folgendes ein:

```
LINSOLVE([X-2*Y+Z=-8,2*X+Y-2*Z=6,5*X-2*Y+Z=-12],
                    [X,Y,Z])
```

um folgende Lösung zu erhalten: `[X=-1,Y=2,Z = -3]`.

Für die Funktion LINSOLVE werden symbolische Ausdrücke verwendet. Die Funktionen REF, rref und RREF werden für die erweiterte Matrix bei der Gauß-Elimination verwendet.

Funktionen REF, rref und RREF

Die obere Dreiecksmatrix, zu der die erweiterte Matrix bei der Vorwärtssubstitution im Rahmen der Gauß-Elimination reduziert wird, wird als Staffelform bezeichnet. Die Funktion REF (Reduce to Echelon Form, zu Staffelform reduzieren) erzeugt eine solche Matrix, wenn die erweiterte Matrix auf Ebene 1 des Stacks vorhanden ist.

Gegeben sei die erweiterte Matrix

$$\mathbf{A}_{aug} = \left[\begin{array}{ccc|c} 1 & -2 & 1 & 0 \\ 2 & 1 & -2 & -3 \\ 5 & -2 & 1 & 12 \end{array}\right].$$

Sie stellt ein lineares Gleichungssystem **A**·**x** = **b** dar, mit

A = `[[1,-2,1],[2,1,-2],[5,-2,1]]`

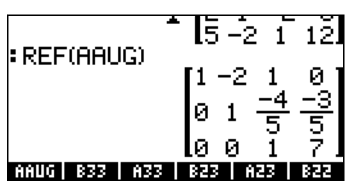
und

b = `[[0],[-3],[12]]`.

Geben Sie die erweiterte Matrix ein und speichern Sie sie im ALG-Modus in der Variablen AAUG:

`[[1,-2,1,0],[2,1,-2,-3][5,-2,1,12]` ▶ `AAUG`

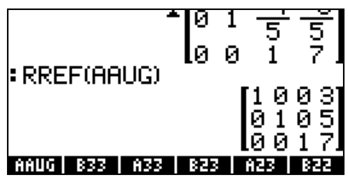
Die Anwendung der Funktion REF erzeugt die Ausgabe:



Das Ergebnis ist die obere Dreieckskoeffizientenmatrix (Staffelform), die aus der Vorwärtssubstitution bei der Gauß-Elimination resultiert.

Die als Ergebnis der Gauß-Jordan-Elimination gebildete Diagonalmatrix wird als zeilenreduzierte Staffelform (bzw. reduzierte Staffelform) bezeichnet.
Funktion RREF (Row-Reduced Echelon Form, zeilenreduzierte Staffelform): Durch Aufruf dieser Funktion wird eine zeilenreduzierte Staffelform erzeugt, so dass die Koeffizientenmatrix zu einer Einheitsmatrix reduziert wird. Die zusätzliche Spalte der erweiterten Matrix enthält die Lösung des Gleichungssystems.

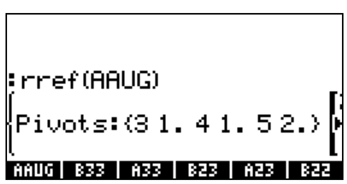
Als Beispiel wird das Ergebnis der Anwendung der Funktion RREF auf die Matrix AAUG im ALG-Modus dargestellt:

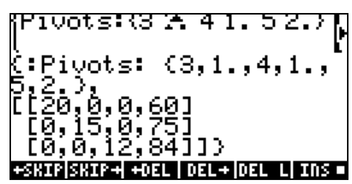


Das Ergebnis ist die durch Gauß-Jordan-Elimination ohne Pivotisierung gebildete, endgültige erweiterte Matrix.

Mit der Funktion rref erhalten Sie eine zeilenreduzierte Staffelform für eine erweiterte Matrix. Diese Funktion erzeugt eine Liste der Pivot-Elemente und eine äquivalente Matrix in zeilenreduzierter Staffelform, so dass die Koeffizientenmatrix zu einer Diagonalmatrix reduziert wird.

Beispielsweise erzeugt die Funktion rref für die Matrix AAUG folgendes Ergebnis:





Die Ausgabe im zweiten Fenster oben erhalten Sie durch Aktivieren des Befehlszeileneditors (drücken Sie ▼). Das Ergebnis enthält die Pivot-Elemente 3, 1, 4, 1, 5 und 2 sowie eine reduzierte Diagonalmatrix.

Funktion SYST2MAT

Mit dieser Funktion wird ein lineares Gleichungssystem in die äquivalente erweiterte Matrix konvertiert. Das folgende Beispiel erhalten Sie über die Hilfefunktion des Taschenrechners:

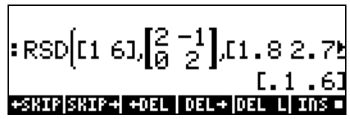
Das Ergebnis ist die dem Gleichungssystem entsprechende erweiterte Matrix:

$$X+Y = 0$$
$$X - Y = 2$$

Restfehler bei Lösungen linearer Gleichungssysteme (Funktion RSD)

Mit der Funktion RSD werden die ReSiDuen bzw. Restfehler bei der Lösung der Matrixgleichung $\mathbf{A}\cdot\mathbf{x}=\mathbf{b}$ berechnet, die ein System von n linearen Gleichungen mit n Unbekannten darstellt. Wir können die Lösung dieses Systems als Lösung der Matrixgleichung $f(\mathbf{x}) = \mathbf{b} - \mathbf{A}\cdot\mathbf{x} = 0$ betrachten. Angenommen, wir erzeugen mit einer numerischen Methode als erste Näherung die Lösung $\mathbf{x}(0)$. Wir berechnen $f(\mathbf{x}(0)) = \mathbf{b} - \mathbf{A}\cdot\mathbf{x}(0) = \mathbf{e} \neq 0$. $\mathbf{e}$ ist somit ein Vektor der Residuen der Funktion für den Vektor $\mathbf{x} = \mathbf{x}(0)$.

Als Argumente der Funktion RSD sind $\mathbf{b}$, $\mathbf{A}$ und $\mathbf{x}(0)$ erforderlich. Der zurückgegebene Vektor lautet $\mathbf{e} = \mathbf{b} - \mathbf{A}\cdot\mathbf{x}(0)$. Wenn beispielsweise $\mathbf{A}$ = `[[2,-1][0,2]]`, $\mathbf{x}(0)$ = `[1.8,2.7]` und $\mathbf{b}$ = `[1,6]`, können wir den Vektor der Residuen wie folgt ermitteln:



Das Ergebnis lautet **e** = **b** - **A**·**x**(0) = `[ 0.1 0.6 ]`.

Hinweis: Wenn wir die Korrektur der Werte von **x**(0) durch den Vektor Δ**x** = **x** – **x** (0) darstellen, können wir für Δ**x** eine neue Matrixgleichung **A**·Δ**x** = **e** erstellen. Durch das Ermitteln von Δ**x** finden wir mit **x** = **x**(0) + Δ**x** die tatsächliche Lösung des ursprünglichen Gleichungssystems.

Eigenwerte und Eigenvektoren

Für eine quadratische Matrix **A** können wir die Eigenwertgleichung **A**·**x** = λ·**x** erstellen, wobei die der Gleichung entsprechenden Werte von λ als Eigenwerte von Matrix **A** bezeichnet werden. Wir können für jeden Wert von λ in der Gleichung Werte von **x** ermitteln, die der Eigenwertgleichung entsprechen. Diese Werte von x werden als Eigenvektoren von Matrix **A** bezeichnet. Die Eigenwertgleichung kann auch als (**A** – λ·**I**)**x** = 0 geschrieben werden.

Diese Gleichung besitzt nur dann eine nicht triviale Lösung, wenn die Matrix (**A** - λ·**I**) singulär ist, d. h., wenn det(**A** - λ·**I**) = 0.

Die letzte Gleichung erzeugt eine algebraische Gleichung mit einem Polynom der Ordnung *n* für eine quadratische Matrix $\mathbf{A}_{n\times n}$. Die resultierende Gleichung wird als charakteristisches Polynom der Matrix **A** bezeichnet. Die Lösung des charakteristischen Polynoms ergibt die Eigenwerte der Matrix.

Der Taschenrechner enthält mehrere Funktionen, mit denen Sie Informationen über Eigenwerte und Eigenvektoren einer quadratischen Matrix erhalten. Einige dieser Funktionen befinden sich im Menü MATRICES/EIGEN, das mit ← MATRICES aktiviert wird.

Funktion PCAR

Mit der Funktion PCAR wird unter Verwendung der Werte der Variablen VX (eine für CAS reservierte Variable, die in der Regel gleich „X" ist) ein charakteristisches Polynom einer quadratischen Matrix erzeugt. Geben Sie beispielsweise im ALG-Modus folgende Matrix ein und ermitteln Sie mit PCAR die charakteristische Gleichung: `[[1,5,-3],[2,-1,4],[3,5,2]]`

Unter Verwendung der Variablen λ zur Darstellung der Eigenwerte ist dieses charakteristische Polynom als λ^3-2λ^2-22λ +21=0 zu interpretieren.

Funktion EGVL

Mit der Funktion EGVL (EiGenVaLues, Eigenwerte) werden die Eigenwerte einer quadratischen Matrix erzeugt. Die Eigenwerte der unten dargestellten Matrix werden z. B. im ALG-Modus mit der Funktion EGVL berechnet:

Eigenwerte λ = [-√10, √10].

Hinweis: In einigen Fällen können Sie möglicherweise keine „exakte“ Lösung für das charakteristische Polynom ermitteln und erhalten bei Verwendung der Funktion EGVL als Ergebnis eine leere Liste. Wenn dieser Fall eintritt, ändern Sie den Berechnungsmodus in CAS in den Näherungsmodus und wiederholen die Berechnung.

Beispielsweise wird bei der folgenden Übung im exakten Modus als Ergebnis eine leere Liste ausgegeben.

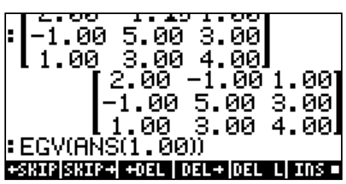
Ändern Sie den Modus in den Näherungsmodus und wiederholen Sie die Eingabe. Sie erhalten folgende Eigenwerte:

`[(1.38,2.22), (1.38,-2.22), (-1.76,0)].`

Funktion EGV

Mit der Funktion EGV (EiGenwerte und EigenVektoren) werden die Eigenwerte und Eigenvektoren einer quadratischen Matrix erzeugt. Die Eigenvektoren werden als die Spalten einer Matrix zurückgegeben, während die entsprechenden Eigenwerte die Komponenten eines Vektors darstellen.

Beispielsweise werden die Eigenvektoren und Eigenwerte der unten aufgeführten Matrix im ALG-Modus durch Anwendung der Funktion EGV ermittelt:



In der Ergebnisliste werden die Eigenwerte als Spalten der Matrix angezeigt. Um die Eigenwerte anzuzeigen, können wir den Befehl GET(ANS(1),2) verwenden, d. h. das zweite Element in der Liste des vorherigen Ergebnisses abrufen. Die Eigenwerte lauten:

Gesamt:

$$\lambda_1 = 0.29,\ \mathbf{x}_1 = [\ 1.00,0.79,–0.91]^T,$$
$$\lambda_2 = 3.16,\ \mathbf{x}_2 = [1.00,-0.51,\ 0.65]^T,$$
$$\lambda_3 = 7.54,\ \mathbf{x}_1 = [-0.03,\ 1.00,\ 0.84]^T.$$

Hinweis: Eine symmetrische Matrix erzeugt alle echten Eigenwerte, und ihre Eigenvektoren sind zueinander orthogonal. Für das gerade erläuterte Beispiel können Sie überprüfen, ob $\mathbf{x}_1 \bullet \mathbf{x}_2 = 0$, $\mathbf{x}_1 \bullet \mathbf{x}_3 = 0$ und $\mathbf{x}_2 \bullet \mathbf{x}_3 = 0$.

Funktion JORDAN

Die Funktion JORDAN ist für die Diagonalisierung oder Jordan-Zerlegung einer Matrix konzipiert. Mit der Funktion JORDAN werden im RPN-Modus für eine quadratische Matrix **A** vier Ausgaben erzeugt:

- Das Minimalpolynom von Matrix **A** (Ebene 4 des Stacks)
- Das charakteristische Polynom von Matrix **A** (Ebene 3 des Stacks)
- Eine Liste der jedem Eigenwert von Matrix **A** entsprechenden Eigenvektoren (Ebene 2 des Stacks)
- Ein Vektor mit den Eigenvektoren von Matrix **A** (Ebene 4 des Stacks)

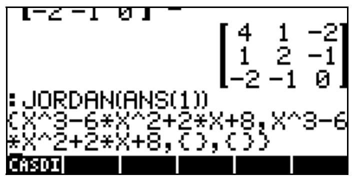
Führen Sie im RPN-Modus beispielsweise diese Übung aus:

```
[[4,1,-2],[1,2,-1],[-2,-1,0]]   JORDAN
```

Die Ausgabe lautet wie folgt:

4: 'X^3+-6*x^2+2*X+8'
3: 'X^3+-6*x^2+2*X+8'
2: { }
1: { }

Dieselbe Übung wird im ALG-Modus wie in den folgenden Bildschirmabbildungen dargestellt:



Funktion MAD

Obwohl diese Funktion nicht im Menü EIGEN zur Verfügung steht, stellt sie auch Informationen über die Eigenwerte einer Matrix bereit. Die Funktion MAD ist im Untermenü MATRICES OPERATIONS (← MATRICES) verfügbar und zum Erzeugen der adjungierten Matrix einer Matrix konzipiert.

Im RPN-Modus erzeugt die Funktion MAD mehrere Eigenschaften einer quadratischen Matrix:

- die Determinante (Ebene 4 des Stacks)
- die formale Inverse (Ebene 3 des Stacks)
- die Matrixkoeffizienten des durch (**x**·**I**-**A**) ·p(**x**)=m(**x**)·**I** definierten Polynoms p(**x**) (Ebene 2 des Stacks)
- das charakteristische Polynom der Matrix (Ebene 1 des Stacks)

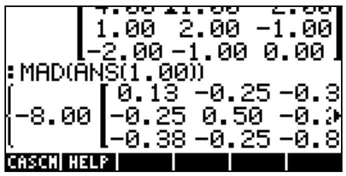
Beachten Sie, dass die Form der Gleichung (**x**·**I**-**A**)·p(**x**)=m(**x**)·**I** der Eigenwertgleichung **A**·**x** = λ·**x** entspricht.

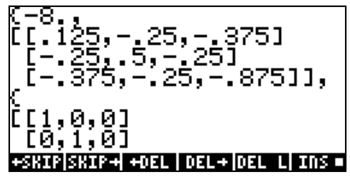
Geben Sie als Übungsbeispiel im RPN-Modus Folgendes ein:

```
[[4,1,-2] [1,2,-1][-2,-1,0]] MAD
```

Das Ergebnis lautet:
4: -8.
3: [[0.13 –0.25 –0.38][-0.25 0.50 –0.25][-0.38 –0.25 –0.88]]
2: {[[1 0 0][0 1 0][0 0 1]] [[-2 1 –2][1 –4 –1][-2 –1 –6] [[-1 2 3][2 –4 2][3 2 7]]}
1: ‘X^3+-6*x^2+2*X+8’

Dasselbe Beispiel wird im ALG-Modus wie folgt angezeigt:





Matrixfaktorisierung

Die Faktorisierung bzw. Zerlegung einer Matrix besteht im Ermitteln von Matrizen, die durch Multiplikation die Ausgangsmatrix ergeben. Wir stellen die Matrixzerlegung durch die Funktionen im Matrixmenü FACT dar. Dieses Menü wird mit ← MATRICES aufgerufen.

Die Funktionen dieses Menüs lauten: LQ, LU, QR,SCHUR, SVD und SVL.

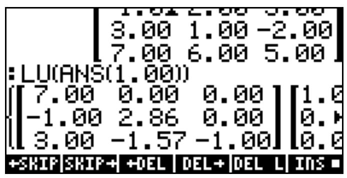
Die Funktion LU

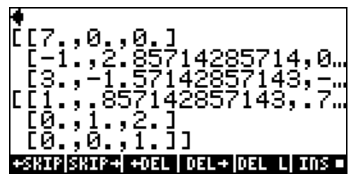
Der Eingabewert für die Funktion LU ist eine quadratische Matrix **A.** Die Ausgabe ist eine untere Dreiecksmatrix **L**, eine obere Dreiecksmatrix **U** und eine Permutationsmatrix **P** auf Ebene 3, 2 bzw. 1 des Stacks. Die Ergebnisse

für **L**, **U** und **P** stimmen mit der Gleichung **P**·**A** = **L**·**U** überein. Beim Aufruf der Funktion LU führt der Taschenrechner mithilfe einer Teilpivotisierung eine LU-Zerlegung von **A** nach dem Crout-Algorithmus durch.
Beispielsweise ergibt die folgende Eingabe im RPN-Modus:
`[[-1,2,5][3,1,-2][7,6,5]] LU`
die Werte:

3:[[7 0 0][-1 2.86 0][3 –1.57 –1]]
2:[[1 0.86 0.71][0 1 2][0 0 1]]
1:[[0 0 1][1 0 0][0 1 0]]

Im ALG-Modus wird dasselbe Beispiel wie folgt angezeigt:





Orthogonalmatrizen und Singulärwertzerlegung

Eine quadratische Matrix ist orthogonal, wenn ihre Spalten Einheitsvektoren darstellen, die zueinander orthogonal sind. Die Matrix **U** = [$\mathbf{v}_1$ $\mathbf{v}_2$ … $\mathbf{v}_n$] mit den Spaltenvektoren $\mathbf{v}_i$, i = 1, 2, …, n und $\mathbf{v}_i \bullet \mathbf{v}_j = \delta_{ij}$, wobei δ_{ij} die Kronecker-Deltafunktion darstellt, ist daher eine Orthogonalmatrix. Aus diesen Bedingungen folgt außerdem, dass $\mathbf{U}\cdot\mathbf{U}^T = \mathbf{I}$.

Die Singulärwertzerlegung (SVD) einer rechteckigen Matrix $\mathbf{A}_{m\times n}$ erfolgt daher durch Bestimmung der Matrizen **U**, **S** und **V**, so dass $\mathbf{A}_{m\times n} = \mathbf{U}_{m\times m} \cdot \mathbf{S}_{m\times n} \cdot \mathbf{V}^T_{n\times n}$, wobei **U** und **V** Orthogonalmatrizen sind und **S** eine Diagonalmatrix ist. Die diagonalen Elemente von **S** werden als <u>Singulärwerte</u> von **A** bezeichnet und sind in der Regel so angeordnet, dass für *i = 1, 2, …, n-1* gilt, dass $s_i \geq s_{i+1}$. Die Spalten [$\mathbf{u}_j$] von **U** und [$\mathbf{v}_j$] von **V** sind die entsprechenden <u>Singulärvektoren</u>.

Funktion SVD

Im RPN-Modus ist der Eingabewert für die Funktion SVD (Singular Value Decomposition, Singulärwertzerlegung) eine Matrix $\mathbf{A}_{n\times m}$. Die Funktion gibt auf den Ebenen 3, 2 bzw. 1 des Stacks die Matrizen $\mathbf{U}_{n\times n}$, $\mathbf{V}_{m\times m}$ und einen

Vektor **s** zurück. Die Dimension des Vektors **s** ist gleich dem geringsten Wert von n bzw. m. Die Matrizen **U** und **V** entsprechen der bereits erläuterten Definition für die Singulärwertzerlegung, während der Vektor **s** die Hauptdiagonale der bereits verwendeten Matrix **S** darstellt.

Beispielsweise ergibt die folgende Eingabe im RPN-Modus:
`[[5,4,-1],[2,-3,5],[7,2,8]] SVD`

3: [[-0.27 0.81 –0.53][-0.37 –0.59 –0.72][-0.89 3.09E-3 0.46]]
2: [[-0.68 –0.14 –0.72][0.42 0.73 –0.54][-0.60 0.67 0.44]]
1: [12.15 6.88 1.42]

Funktion SVL
Die Funktion SVL (Singular VaLues, Singulärwerte) gibt die Singulärwerte einer Matrix $\mathbf{A}_{n\times m}$ als Vektor **s** zurück, dessen Dimension gleich dem geringsten Wert von n bzw. m ist. Beispielsweise ergibt
`[[5,4,-1],[2,-3,5],[7,2,8]] SVL` im RPN-Modus
[12.15 6.88 1.42].

Funktion SCHUR
Im RPN-Modus erzeugt die Funktion SCHUR die *Schur-Zerlegung* einer quadratischen Matrix **A** und ergibt die Matrizen **Q** und **T** auf Ebene 2 bzw. 1 des Stacks, so dass $\mathbf{A} = \mathbf{Q}\cdot\mathbf{T}\cdot\mathbf{Q}^T$, wobei **Q** eine Orthogonalmatrix und **T** eine Dreiecksmatrix ist. Beispielsweise ergibt

`[[2,3,-1][5,4,-2][7,5,4]] SCHUR`

im RPN-Modus folgende Ausgabe:
2: [[0.66 –0.29 –0.70][-0.73 –0.01 –0.68][-0.19 –0.96 0.21]]
1: [[-1.03 1.02 3.86][0 5.52 8.23][0 –1.82 5.52]]

Funktion LQ
Die Funktion LQ erzeugt die *LQ-Faktorisierung* einer Matrix $\mathbf{A}_{n\times m}$ und gibt Ebene 3, 2 bzw. 1 des Stacks eine untere Trapezmatrix $\mathbf{L}_{n\times m}$, eine Orthogonalmatrix $\mathbf{Q}_{m\times m}$ und eine Permutationsmatrix $\mathbf{P}_{n\times n}$ zurück. Für die Matrizen **A**, **L**, **Q** und **P** gilt $\mathbf{P}\cdot\mathbf{A} = \mathbf{L}\cdot\mathbf{Q}$. (Eine aus einer n×m-Matrix gebildete Trapezmatrix entspricht einer aus einer n×n-Matrix gebildeten Dreiecksmatrix.) Beispielsweise erzeugt

`[[ 1, -2, 1][ 2, 1, -2][ 5, -2, 1]] LQ`

die Werte
3: [[-5.48 0 0][-1.10 –2.79 0][-1.83 1.43 0.78]]
2: [[-0.27 0.81 –0.18][-0.36 –0.50 –0.79][-0.20 –0.78 –0.59]]
1: [[0 0 1][0 1 0][1 0 0]]

Funktion QR

Die Funktion QR erzeugt im RPN-Modus die *QR-Faktorisierung* einer Matrix $\mathbf{A}_{n\times m}$ und gibt auf Ebene 3, 2 bzw. 1 des Stacks eine Orthogonalmatrix $\mathbf{Q}_{n\times n}$, eine obere Trapezmatrix $\mathbf{R}_{n\times m}$ und eine Permutationsmatrix $\mathbf{P}_{m\times m}$ zurück. Für die Matrizen **A**, **P**, **Q** und **R** gilt **A**·**P** = **Q**·**R**. Beispielsweise erzeugt
`[[ 1,-2,1][ 2,1,-2][ 5,-2,1]] QR`
die Werte
3: [[-0.18 0.39 0.90][-0.37 –0.88 0.30][-0.91 0.28 –0.30]]
2: [[-5.48 –0.37 1.83][0 2.42 –2.20][0 0 –0.90]]
1: [[1 0 0][0 0 1][0 1 0]]

Hinweis: Über die Hilfefunktion des Taschenrechners erhalten Sie Beispiele und Definitionen für sämtliche Funktionen dieses Menüs. Führen Sie diese Übungen im ALG-Modus aus, um die Ergebnisse in diesem Modus anzuzeigen.

Quadratische Formen einer Matrix

Die _quadratische Form_ einer quadratischen Matrix **A** ist ein aus $\mathbf{x}\cdot\mathbf{A}\cdot\mathbf{x}^T$ gebildetes Polynom. Für **A** = [[2,1,–1][5,4,2][3,5,–1]] und **x** = $[X\ Y\ Z]^T$ wird z. B. die entsprechende quadratische Form wie folgt berechnet:

$$\mathbf{x}\cdot\mathbf{A}\cdot\mathbf{x}^T = \begin{bmatrix} X & Y & Z \end{bmatrix}\cdot\begin{bmatrix} 2 & 1 & -1 \\ 5 & 4 & 2 \\ 3 & 5 & -1 \end{bmatrix}\cdot\begin{bmatrix} X \\ Y \\ Z \end{bmatrix}$$

$$= \begin{bmatrix} X & Y & Z \end{bmatrix}\cdot\begin{bmatrix} 2X+Y-Z \\ 5X+4Y+2Z \\ 3X+5Y-Z \end{bmatrix}$$

Endergebnis: $\mathbf{x}\cdot\mathbf{A}\cdot\mathbf{x}^T = 2X^2+4Y^2-Z^2+6XY+2XZ+7ZY$

Das Menü QUADF

Der Taschenrechner HP 49 G enthält das Menü QUADF für Operationen mit QUADratischen Formen. Das Menü QUADF wird mit ← MATRICES aufgerufen.

Dieses Menü enthält die Funktionen AXQ, CHOLESKY, GAUSS, QXA und SYLVESTER.

Funktion AXQ

Die Funktion AXQ erzeugt im RPN-Modus unter Verwendung von *n* Variablen eines Vektors auf Ebene 1 des Stacks die einer Matrix $\mathbf{A}_{n\times n}$ auf Ebene 2 des Stacks entsprechende quadratische Form. Die Funktion gibt die quadratische Form auf Ebene 1 des Stacks und den Vektor der Variablen auf Ebene 1 des Stacks zurück. Beispielsweise ergibt

```
[[2,1,-1],[5,4,2],[3,5,-1]] ENTER
['X','Y','Z'] ENTER AXQ
```

folgende Werte:
2: '2*X^2+(6*Y+2*Z)*X+4*Y^2+7*Z*y-Z^2'
1: ['X' 'Y' 'Z']

Funktion QXA

Die Argumente für die Funktion QXA sind eine quadratische Form auf Ebene 2 des Stacks und ein Vektor von Variablen auf Ebene 1 des Stacks. Die Funktion gibt auf Ebene 2 des Stacks die quadratische Matrix **A** zurück, von der die quadratische Form abgeleitet wurde, und auf Ebene 1 des Stacks die Liste der Variablen. Beispielsweise ergibt

```
'X^2+Y^2-Z^2+4*X*Y-16*X*Z' ENTER
['X','Y','Z'] ENTER QXA
```

folgende Werte:

2: [[1 2 –8][2 1 0][-8 0 –1]]
1: ['X' 'Y' 'Z']

Diagonale Darstellung einer quadratischen Form

Für eine symmetrische quadratische Matrix **A** kann die Matrix **A** „diagonalisiert" werden, indem eine Orthogonalmatrix **P** ermittelt wird, für die gilt: $\mathbf{P}^T\cdot\mathbf{A}\cdot\mathbf{P} = \mathbf{D}$, wobei **D** eine Diagonalmatrix ist. Wenn $Q = \mathbf{x}\cdot\mathbf{A}\cdot\mathbf{x}^T$ eine auf **A** basierende quadratische Form ist, kann die quadratische Form Q so dargestellt werden, dass sie mit $Q = \mathbf{x}\cdot\mathbf{A}\cdot\mathbf{x}^T = (\mathbf{P}\cdot\mathbf{y})\cdot\mathbf{A}\cdot(\mathbf{P}\cdot\mathbf{y})^T = \mathbf{y}\cdot(\mathbf{P}^T\cdot\mathbf{A}\cdot\mathbf{P})\cdot\mathbf{y}^T = \mathbf{y}\cdot\mathbf{D}\cdot\mathbf{y}^T$ nur quadratische Ausdrücke einer Variablen **y** enthält, so dass $\mathbf{x} = \mathbf{P}\cdot\mathbf{y}$.

Funktion SYLVESTER

Die Funktion SYLVESTER nimmt eine symmetrische quadratische Matrix **A** als Argument an und gibt einen Vektor zurück, der die diagonalen Größen einer Diagonalmatrix **D** enthält, sowie eine Matrix **P**, so dass $\mathbf{P}^T\cdot\mathbf{A}\cdot\mathbf{P} = \mathbf{D}$.
Beispielsweise ergibt

```
[[2,1,-1],[1,4,2],[-1,2,-1]] SYLVESTER
```

die Werte
2: [1/2 2/7 -23/7]
1: [[2 1 –1][0 7/2 5/2][0 0 1]]

Funktion GAUSS

Die Funktion GAUSS gibt die diagonale Darstellung einer quadratischen Form $Q = \mathbf{x}\cdot\mathbf{A}\cdot\mathbf{x}^T$ zurück und nimmt als Argumente die quadratische Form auf Ebene 2 des Stacks und den Vektor der Variablen auf Ebene 1 des Stacks an. Der Aufruf dieser Funktion führt zu folgenden Ergebnissen:

- Ein Feld von Koeffizienten, die die diagonalen Größen von **D** darstellen (Ebene 4 des Stacks)
- Eine Matrix **P**, so dass $\mathbf{A} = \mathbf{P}^T\cdot\mathbf{D}\cdot\mathbf{P}$ (Ebene 3 des Stacks)
- Die diagonalisierte quadratische Form (Ebene 2 des Stacks)
- Die Liste der Variablen (Ebene 1 des Stacks)

Beispielsweise erzeugt

```
'X^2+Y^2-Z^2+4*X*Y-16*X*Z' ENTER
['X','Y','Z'] ENTER GAUSS
```

folgende Werte:

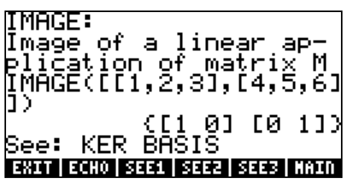
4: [1 –0.333 20.333]
3: [[1 2 –8][0 –3 16][0 0 1]]
2: '61/3*Z^2+ -1/3*(16*Z+-3*Y)^2+(-8*z+2*Y+X)^2'
1: ['X' 'Y' 'Z']

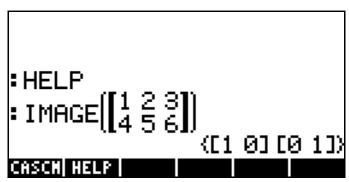
LINEAR APPLICATIONS

Das Menü LINEAR APPLICATIONS wird über ⬅ MATRICES aufgerufen.

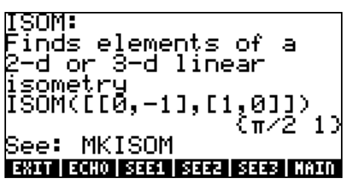
Unten sind die Informationen über die Funktionen dieses Menüs dargestellt, die Sie mit der Hilfefunktion des Taschenrechners aufrufen können. Die Abbildungen stellen den entsprechenden Eintrag der Hilfefunktion und die zugehörigen Beispiele dar.

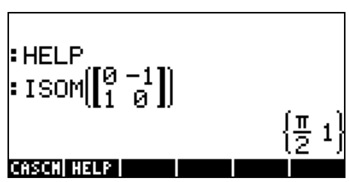
Funktion IMAGE



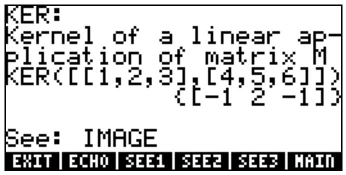


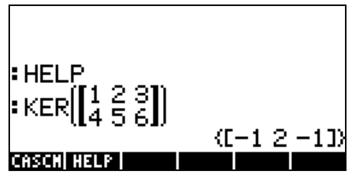
Funktion ISOM



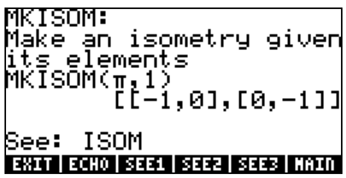


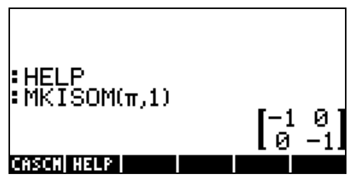
Funktion KER





Funktion MKISOM



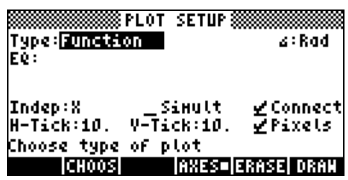


Kapitel 12
Graphiken

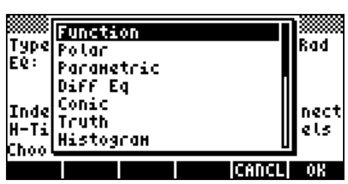
In diesem Kapitel führen wir einige der Graphikfähigkeiten des Rechners ein. Wir stellen Graphiken von Funktionen in den kartesischen Koordinaten und in den polaren Koordinaten, in den parametrischen Plots, in den Graphiken von conics, in den Stabplots, in den scatterplots und in einer Vielzahl der dreidimensionalen Diagramme dar.

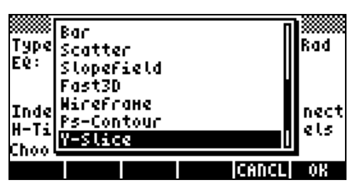
Diagrammwahlen im Rechner

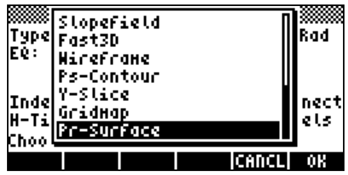
Um die Liste der graphischen Formate zugänglich zu machen, die im Rechner vorhanden sind, benutzen Sie die Tastenreihen ← 2D/3D (F4) Beachten Sie bitte daß wenn Sie den RPN Modus verwenden diese zwei Taste gleichzeitig betätigt werden müssen um irgendwelche der Diagrammfunktionen zu aktivieren. Nachdem er die Funktion 2D/3D aktiviert hat, produziert der Rechner das PLOT-SETUP-Fenster, daß das TYPE Feld einschließt, wie unten gezeigt wird.



Recht vor dem TYPE Feld, werden Sie am wahrscheinlichsten sehen daß die Wahlfunktion hervorgehoben wurde. Dieses ist die Rückstellungstype des Diagramms für den Rechner. Um die Liste der vorhandenen Diagrammarten zu sehen, betätigen Sie den weichen markierten MenüTaste CHOOS. Dieses produziert ein drop down Menü mit den folgenden Optionen (verwenden Sie die hohen und Untenpfeil Tasten, um alle Optionen zu sehen):







Diese Diagrammwahlen werden zunächst kurz beschrieben.

Function : für Gleichungen der Form y = f(x) in den flachen kartesischen Koordinaten
Polar : für Gleichungen von von r = f(q) in den polaren Koordinaten in der Fläche
Parametric : für das Plotten der Gleichungen der Form x = x(t), y = y(t) in der Fläche
Diff Eq : für das Plotten der numerischen Lösung einer linearen Differentialgleichung
Conic : für das Plotten der konischen Gleichungen (Kreise, Ellipses, Hyperbeln, Parabeln)
Truth : für das Plotten von von Verschiedenheiten in der Fläche
Histogram : für das Plotten der Frequenzhistogramme (statistische Anwendungen)
Bar : für das Plotten der einfachen Balkendiagramme
Scatter : für das Plotten der Streuungplots der getrennten Modems (statistische Anwendungen)
Slopefield : für das Plotten der Spuren der Steigungen eines Funktion f(x, y) = 0.
Fast3D : für das Plotten der gebogenen Oberflächen im Raum
Wireframe : für das Plotten der gebogenen Oberflächen im Raum, der Leitung Rahmenrasterfelder zeigt
Ps-Contour : für das Plotten der Formplots der Oberflächen
Y- Slice e : für das Plotten einer schneidenen Ansicht eines Funktion f(x, y).
Gridmap : für das Plotten der Spuren des realen und Imaginärteils einer komplizierten Funktion
Pr-Surface : für die parametrischen Oberflächen gegeben durch x = x(u, V), y = y(u, V), z = z(u, V).

Einen Ausdruck der Form y = f(x) plotten

In diesem Abschnitt stellen wir ein Beispiel eines Plots einer Funktion der Form y = f(x) dar. Um dem Plot fortzufahren, erst mal das Variable x bereinigen, wenn es im aktuellen Verzeichnis definiert wird (x ist das unabhängige Variable in der PLOT-Eigenschaft des Rechners, folglich möchten Sie das nicht vorbestimmen lassen). Machen Sie ein Unterverzeichnis, das ' TPLOT ' genannt oder anderen sinnvollen Namen (für Testplot), um die folgende Übung durchzuführen.

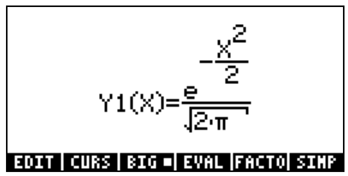
Als Beispiel lassen wir die Funktion plotten,

$$f(x) = \frac{1}{\sqrt{2\pi}} \exp(-\frac{x^2}{2})$$

- zuerst, tragen Sie das PLOT-SETUP-Klima ein, dabei ↰ 2D/3D betätigend.Überprüfen Sie, ob die Wahlfunktion als der TYPE vorgewählt wird und daß 'X' als die unabhängige Variable (INDEP) wird vorgewählt. · Betätigen Sie NXT OK um zur normalen Rechneranzeige zurückzugehen. das PLOT-SETUP-Fenster sollte diesem ähnlich sein:
-

- **Anmerkung**: beachten Sie, daß ein neues Variables, benannt PPAR, oben in Ihren weichen MenüTastemarken zeigt. Dieses für Plot-Parameter steht. Um sein Inhalt zu sehen, betätigen Sie ↱ PPAR. Eine genau geschilderte Erklärung des Inhalts von PPAR wird später in diesem Kapitel zur Verfügung gestellt. Betätigen Sie ◄ um diese Linie vom Stapel fallenzulassen

- Tragen Sie das PLOT-Klima ein, indem Sie gleichzeitig ← Y= betätigen wenn Sie im RPN Modus sind). Betätigen Sie ADD wobei Sie in den Gleichungsverfasser kommen. Sie werden aufgefordert, die rechte Seite einer Gleichung Y1(x) =▪. zu füllen Tippen Sie die Funktion die geplottet werden muss, damit der Gleichungsverfasser das folgende zeigt
-



- Betätigen Sie ENTER um in dem PLOT SETUP Fenster zurückzukehren. Der Ausdruck 'Y1(X) = EXP(-X^2/2)/√(2*π)' wird hervorgehoben sein. Betätigen Sie NXT OK um zur normalen Rechneranzeige zurückzugehen.

Anmerkung: Zwei neue Variablen zeigen sich in Ihren WeichmenüTastemarken, nämlich EQ und Y1. Um der Inhalt von EQ zu sehen, benützen Sie → EQ. Der Inhalt van EQ is einfach der Funktionsnamen 'Y1(X)'. Das Variable EQ wird benützt von dem Rechner um die Gleichung(en) zu speichern die geplottet werden müssen.

Um der Inhalt von Y1 zu sehen, benützen Sie → Y1. Sie erhalten die Funktion Y1(X) definiert als das Programm:

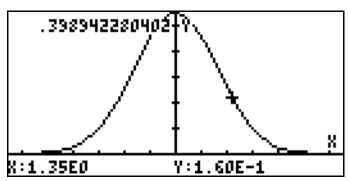
```
<< →X 'EXP(-X^2/2)/√(2*π)' >>.
```

Betätigen Sie zweimal ◄ um den Inhalt des Stapels fallenzulassen.

- Tragen Sie das PLOT WINDOW-Klima ein, indem Sie gleichzeitig ← Y= betätigen wenn Sie im RPN Modus sind. Benützen Sie eine Strecke von –4 bis 4 für H-VIEW, danach betätigen Sie AUTO um

V-VIEW automatisch zu generieren. Der Schirm vom PLOT WINDOW-Fenster sieht so aus:

- Plotten Sie das Diagramm: ERASE DRAW (bitte warten bis den Rechner das Diagramm fertig hat)
- Um Marken zu sehen: EDIT NXT LABEL MENU
- Um das erste Graphikmenü zurückzugewinnen: NXT NXT PICT
- Um Die Kurve zu verfolgen: TRACE (X,Y) . Verwenden Sie dann die Recht- und LinkpfeilTaste (◀ ▶) um über die Kurve zu bewegen. Die Koordinaten der Punkten, die Sie verfolgen, werden am unteren Bildschirmrand gezeigt. Prüfen Sie daß für x = 1.05 , y = 0.231. Prüfen Sie auch daß für x = -1.48 , y = 0.134. Hier ist ein Bild vom Diagramm im Verfolgungsmodus:



- Um das Menü zurückzugewinnen und zur PLOT WINDOW-Klima zurückzugehen, betätigen Sie NXT CANCL, und dann NXT OK.

Einige nützlichen PLOT-Betriebe für FUNCTION Plots

Um diese PLOT-Optionen zu besprechen, ändern wir die Funktion um sie zu zwingen um einige reale Wurzeln zu haben. (Da die gegenwärtige Kurve tatal über der X Mittellinie enthalten wird, hat sie keine realen Wurzeln). Betätigen Sie ↱ Y1 um den Inhalt der Funktion Y1 in dem Stapel aufzuzeignen: << →X ‘EXP(-X^2/2)/ √(2*π) ’ >>. Um dieser Ausdruck zu redigieren, benützen Sie:

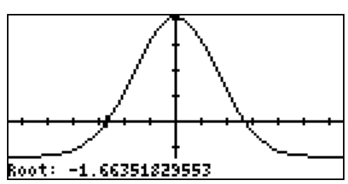
▼	Öffnet der Linie-Editor
↱ ▼	Verschiebt der Cursor auf das Ende der Linie
◀ ◀ ◀ − 0 • 1	Ändert der Ausdruck
ENTER	Geht zur Rechneranzeige zurück

Zunächst speichern Sie den geänderten Ausdruck in dem Variable y, indem Sie ↰ Y1 verwenden wenn Sie im RPN Modus oder ↰ ANS STO▸ Y1 im ALG Modus sind .

Wir können jetzt die Funktion plotten, $f(x) = \frac{1}{\sqrt{2\pi}}\exp(-\frac{x^2}{2}) - 0.1$

Tragen Sie das PLOT WINDOW-Klima ein, indem Sie gleichzeitig ↰ WIN betätigen wenn Sie im RPN Modus sind. Benützen Sie eine Strecke von –4 bis 4 für H-VIEW, danach betätigen Sie ▼ AUTO um V-VIEW automatisch zu generieren. Um das Diagramm zu plotten, betätigen Sie ERASE DRAW

- Als das Diagramm geplottet ist, betätigen Sie FCN um in das Funtkionmenü zu geraten. Mit diesem Menü, können Sie zusätzliche Informationen über den Plot einholen wie schneiden mit dem X-axis, den Wurzeln, den Steigungen der Tangentelinie, Bereich unter der Kurve, usw..
- Z.B. um die Wurzel auf der linken Seite der Kurve zu finden, verschieben Sie den Cursor nah an diesem Punkt, und betätigen Sie ROOT. Sie bekommen das Resultat: ROOT: -1.6635…. Betätigen Sie NXT um zurückzugehen im Menü. Dies ist das Resultat von ROOT in dem gegenwärtigen Plot
-



- Wenn Sie den Cursor richtung der rechten Seite vom Kurve bewegen, weil Sie den RechtpfeilTaste (▶) betätigen, und betätigen Sie ROOT, wird das Resultat jetzt ROOT sein: 1.6635... Der Rechner zeigte daß der Wurzel gefunden wurde mit *SIGN REVERSAL,* bevor sie gezeigt wurde. Betätigen Sie NXT um zurückzugehen im Menü.
- Das betätigen von ISECT gibt Ihnen den Durchschnitt der Kurve mit dem X-axis, der im Wesentlichen die Wurzel ist . Setzen Sie den Cursor genau an der Wurzel und betätigen Sie ISECT. Sie bekommen der gleichen Bericht als vorher, nämlich *SIGN REVERSAL* bevor der Resultat I-SECT erhalten wird: 1.6635…. Die Funktion ISECT soll den Durchschnitt aller möglicher zwei Kurven feststellen, die zur Position des Cursors am nähsten sind. In diesem Fall wo nur eine Kurve beteiligt ist, nämlich Y1(X) , ist der gesuchte Durchschnitt der von f(x) mit dem X-axis, jedoch müssen Sie das Cursor-Recht an der Wurzel setzen, um das gleiche Resultat zu produzieren. Betätigen Sie NXT um zurückzugehen im Menü.
- Setzen Sie den Cursor auf die Kurve an irgendeinem Punkt und betätigen Sie SLOPE um den Wert der Steigung an diesem Punkt zu erhalten. Z.B. an der negativen Wurzel , SLOPE: 0.16670…. Betätigen Sie NXT um zurückzugehen im Menü.
- Um den höchsten Punkt in der Kurve festzustellen, setzen Sie den Cursor nahe dem Gipfel und betätigen Sie EXTR Das Resultat ist EXTRM: 0.. Betätigen Sie NXT um zurückzugehen im Menü.
- Andere Tasten die im ersten Menü vorhanden sind,sind AREA um den Bereich unter der Kurve zu errechnen und SHADE einen Bereich unter der Kurve zu schattieren. Betätigen Sie NXT um mehrere Optionen zu sehen. Das zweite Menü umfaßt einen Taste genannt VIEW daß blitzt für einige Sekunden, in welche die Gleichung plottete. Betätigen Sie VIEW.. Wechselweise können Sie die Taste NEXQ betätigen (folgende Gleichung) um den Namen der Funktion Y1(x) zu sehen. Betätigen Sie NXT um zurückzugehen im Menü.
- Die Taste F(X) gibt den Wert von f(x) entsprechend der Cursor-Position Stellen Sie der Cursor irgendwo im Kurve und betätigen Sie F(X). Der Wert wird in der untereren linken Ecke der Anzeige gezeigt. Betätigen Sie NXT um zurückzugehen im Menü.

- Legen Sie den Cursor in irgendeinen gegebenen Punkt der Flugbahn und betätigen Sie TANL, um die Gleichung der Tangentelinie zur Kurve an diesem Punkt zu erhalten   Die Gleichung wird in der untereren linken Ecke der Anzeige gezeigt.   Betätigen Sie NXT um zurückzugehen im Menü.
- Wenn Sie F' betätigten, wird der Rechner die abgeleitete Funktion f'(x) = df/dx plotten, und auch die originellen Funktion f(x). Beobachten Sie daß die zwei Kurven abschneiden bei zwei Punkten. Verschieben Sie den Cursor nahe dem linken Abschnittpunkt und betätigen Sie FCN ISECT, um I-SECT zu erhalten: (-0.6834…,0.21585). Betätigen Sie NXT um zurückzugehen im Menü.
- Verschieben Sie den Cursor nahe dem linken Abschnittpunkt und betätigen Sie PICT (oder NXT PICT).
- Betätigen Sie CANCL um in dem PLOT WINDOW-Klima zurückzukehren.   Dann betätigen Sie NXT OK um zur normalen Rechneranzeige zurückzugehen.

Anmerkung: der Stapel zeigt alle Diagrammarbeiten, die durchgeführt werden, richtig gekennzeichnet.

- Tragen Sie das PLOT-Klima ein, indem Sie, gleichzeitig im RPN Modus ← Y= betätigen. Beachten Sie daß das hervorgehoben Feld im PLOT-Klima jetzt der Ableitung von Y1(X) enthält. · Betätigen Sie NXT OK um zur normalen Rechneranzeige zurückzugehen.
- Betätigen Sie → EQ um der Inhalt der EQ zu prüfen. Sie werden sehen, daß er eine Liste statt ein einzelner Ausdruck enthält. Die Liste hat als als Elemente ein Ausdruck für die Ableitung von Y1(X) und von Y1(X) selbst.   Ursprünglich enthielt EQ nur Y1(x).   Nachdem wir F' im FCN Klima betätigten, fügte der Rechner automatisch die Ableitung von Y1(x) der Liste von Gleichungen in EQ hinzu .

Ein Diagramm speichern für zukünftigen Gebrauch

Wenn Sie Ihr Diagramm zu einer Variable speichern möchten, müssen Sie in dem PICTURE-Klima geraten durch das Betätigen von ◄.   Danach betätigen Sie EDIT NXT NXT PICT→. Dieses fängt die gegenwärtige Abbildung in einen

Graphikgegenstand. Um zurückzukehren zum Stapel, betätigen Sie PICT CANCL.

In Niveau 1 des Stapels sehen Sie Graphiken beschrieben als `131 × 64`. Dieses kann in einen variablen Namen. sagen wir PIC1gespeichert werden.

Um Ihre Abbildung wieder anzuzeigen, erinnern Sie sich an den Inhalt von Variablen PIC1 zum Stapel. Der Stapel zeigt die Linie: Graphik `131 × 64`. Um das Diagramm zu sehen, tragen Sie das PICTURE- Klima ein, indem Sie sich betätigen ◀.

Löschen Sie die gegenwärtige Abbildung, EDIT NXT ERASE.

Verschieben Sie den Cursor auf die obere linke Ecke der Anzeige, indem Sie die ◀ und ▲ Taste verwenden.

Die Abbildung in Niveau 1 der Stapelpresse NXT REPL anzeigen.

Um zurückzukehren zum normalen Rechners, betätigen Sie PICT CANCL.

Anmerkung: Um sparsam zu sein mit dem Druckenraum umfassen wir nicht mehr Diagramme die nach die Anweisungen in diesem Kapitel produziert werden. Der Benutzer wird eingeladen, jene Graphiken selbst zu produzieren.

Graphiken von Transcendentalfunktionen

In diesem Abschnitt benutzen wir einige der Graphikeigenschaften des Rechners, um das typische Verhalten der Funktionen von natürlichen Maschinenbordbuch-, exponentialen, trigonometrischen und Hyperbelfunktionen zu zeigen. Sie sehen keine Diagramme in diesem Kapitel, denn ich wünsche Sie werden sie in Ihrem Rechner sehen.

Diagramm von ln(X)

Betätigen Sie, gleichzeitig wenn in RPN Modus die Umschalttaste auf der linken Seite ↰ und die Taste the 2D/3D (F4)um das PLOT SETUP Fenster zu produzieren. Das Feld markiert als **Type** wird hervorgehoben. Wenn die Optionsfunktion nicht bereits vorgewählt ist, betätigen Sie die frei belegbare Funktionstaste, die markiert wird als CHOOS, verwenden Sie die Auf- und Ab-Tasten, um `Function` vorzuwählen, und betätigen Sie OK um die Vorwähler durchzuführen. Prüfen Sie ob das Feld markiert ist als Indep: das Variable 'X' enthält. Wenn das nicht so ist, betätigen Sie die unten Pfeiltaste zweimal, bis das `Indep` Feld hervorgehoben ist, betätigen Sie danach die frei belegbare Funktionstaste, die beschriftet EDIT wird, und ändern Sie den Wert der unabhängigen Variable, um 'X' zu lesen. Betätigen Sie OK wenn fertig. Betätigen Sie NXT OK um zur normalen Rechneranzeige zurückzugehen.

Betätigen Sie, als Nächstes verändern wir die Größe des Plotfensters. Zunächst, gleichzeitig wenn in RPN Modus die Umschalttaste auf der linken Seite ↰ und die Taste Y= (F1) um das PLOT-FUNCTION-Fenster zu produzieren. Wenn es irgendeine Gleichung gibt, die in diesem Fenster hervorgehoben wird, betätigen Sie DEL wie gebraucht um das Fenster vollständig zu löschen. Wenn das PLOT-FUNCTION-Fenster leer ist, erhalten Sie eine sofortige Anzeige, die liest: **No Equ., Press ADD**. Betätigen Sie die frei belegbare Funktionstaste markiert ADD . Dieses ruft den Gleichungsverfasser an mit dem Ausdruck Y1(X)=◄ . Tippen Sie LN(X). Betätigen Sie ENTER um in dem PLOT-FUNCTION-Fenster zurückzukehren. · Betätigen Sie NXT OK um zur normalen Rechneranzeige zurückzugehen.

Der folgende Schritt ist zu betätigen, gleichzeitig wenn im RPN Modus,die Umschalttaste auf der linken Seite ↰ und die Taste WIN (F2) um das PLOT-WINDOW-FUNCTION-Fenster zu produzieren. Am wahrscheinlichsten, zeigt die Anzeige die horizontalen (H-VIEW) und vertikalen (`V-View`) Strecken wie: H-VIEW: -6,5 6,5, V-VIEW: -3.1 3.2

Diese sind die Anfangs-Werte für die X und die Y-strecke beziehungsweise des gegenwärtigen An zeigefensters der Graphik. Ändern Sie den Wert von H-VIEW um zu lesen: H-VIEW: -1 10 durch das Verwenden von 1 +/- OK 1 0 OK. Zunächst betätigen Sie die frei belegbare markierte Funktionstaste AUTO um den Rechner die entsprechende vertikale Strecke feststellen zu lassen. Nachdem ein Paar Sekunden vorbei sind, wird diese Strecke im PLOT WINDOW-FUNCTION- Fenster gezeigt. An diesem

Punkt sind wir bereit, das Diagramm zu produzieren von ln(X). Betätigen Sie ERASE DRAW um die natürlichen Logarithmusfunktion zu plotten.

Um Marken in dem Diagramm hinzufügen betätigen Sie EDIT NXT LABEL. Betätigen Sie MENU um die Menümarken zu entfernen und erhalten Sie eine volle Ansicht vom Diagramm. Betätigen Sie NXT um zurückzugehen im Menü. Betätigen Sie NXT PICT um das ursprüngliche graphische Menü zurückzugewinnen. .

Um die Koordinaten der Punkte auf der Kurve festzustellen betätigen Sie TRACE (der Cursor bewegt auf die Kurve an einem Punkt, der nahe der Mitte der horizontalen Strecke gelegen ist). Zunächts, betätigen Sie (X,Y) um die Koordinaten des gegenwärtigen Positionsanzeigers zu sehen. Diese Koordinaten werden am unteren Bildschirmrand gezeigt. Verwenden Sie die Recht- und LinkpfeilTaste, den Cursor der Kurve entlang zu verschieben. Da Sie den Cursor entlang der Kurve verschieben, werden die Koordinaten der Kurve am unteren Bildschirmrand angezeigt. Prüfen Sie ob wenn Y:1.00E0, X:2.72E0. Dieses ist der Punkt (*e,1*), seit *ln(e) = 1*. Betätigen Sie NXT um das Graphikmenü zurückzugewinnen.

Zunächst finden wir den Durchschnitt der Kurve mit dem X-axis, indem wir FCN ROOT betätigen. Der Rechner bringt die Wert `Root` zurück: 1, daß *ln(1) = 0*. bestätigt. Betätigen Sie NXT NXT PICT CANCL um zum PLOT-WINDOW-FUNCTION zurückzugehen. Betätigen Sie ENTER um zur normalen Rechneranzeige zurückzugehen. Sie werden beachten, daß die Wurzel, die im Graphikklima gefunden wurde, zum Rechnerstapel kopiert wurde.

Anmerkung: Wenn Sie VAR betätigen, zeigt Ihre Variablenliste die neuen Variablen, die genannt X und Y1 werden. Betätigen Sie ↱ Y1 um der Inhalt dieses Variablen zu sehen.. Sie erhalten das Programm << → X 'LN(X)' >> , daß Sie als das Programm erkennen, daß aus dem Definieren der Funktion 'Y1(X) = LN(X)' resultieren kann durch das Verwenden von ↰ DEF . Dieses ist im Allgemeinen, was geschieht, wenn Sie ADD eine Funktion im PLOT-FUNCTION Fenster (das Fenster daß resultierte vom Betätigen : ↰ Y= , gleichzeitig wenn in RPN Modus), d.h. die Funktion wird definiert und an Ihren Variableliste hinzugefügt.

Betätigen Sie ⓡ X um der Inhalt dieses Variablen zu sehen. Ein Wert von 10,275 wird in den Stapel gelegt. Dieser Wert wird durch unsere Vorwähler für die horizontale Anzeigestrecke festgestellt. Wir wählten eine Strecke zwischen -1 und 10 für X vor. Um das Diagramm zu produzieren, erzeugt der Rechner Werte zwischen den Strecke-begrenzungen mit einer konstanten Stufensprung, und die Werte speichernd, die erzeugt werden, eins nach dem anderen, im Variablen X wie das Diagramm gezeichnet wird. Für die horizontale Strecke (–1,10), scheint die verwendete Stufensprung, 0,275 zu sein. Wenn der Wert von X größer wird als der Maximalwert in der Strecke (in diesem Fall, wenn X = 10,275), stoppt die Zeichnung des Diagramms. Der letzte Wert von X für die Graphik in Erwägung wird in variablem X gehalten. Löschen Sie X und Y1 bevor dem Fortfahren.

Diagramm von der Exponentialfunktion

Zuerst laden Sie die Funktion *exp(X),* gleichzeitig betätigend wenn in RPN Modus, die Umschalttaste auf der linken Seite ⓛ und die Taste Y= (EEX) um das PLOT- FUNCTION-Fenster hervorzu rufen. Betätigen Sie DEL um die Funktion LN(X) zu entfernen, wenn Sie Y1 nicht entfernt haben wie in der vorhergehenden Anmerkung vorgeschlagen wurde. Betätigen Sie ADD und tippen Sie ⓛ e^x ALPHA X ENTER um EXP(X) einzutragen und zum Plotten-Funktion Fenster zurückzugehen. · Betätigen Sie NXT OK um zur normalen Rechneranzeige zurückzugehen.

Betätigen Sie, gleichzeitig wenn in RPN Modus die Umschalttaste auf der linken Seite ⓛ und die Taste WIN (F2) um das PLOT WINDOW-FUNCTION-Fenster zu produzieren. Ändern Sie den Wert von H-VIEW um zu lesen: H-VIEW: -8 2

durch das Verwenden von 8 +/- OK 2 OK. Zunächst betätigen Sie AUTO.. Nachdem die vertikale Strecke errechnet ist, betätigen Sie ERASE DRAW um die Exponentialfunktion zu plotten.

Um Marken in dem Diagramm hinzufügen betätigen Sie EDIT NXT LABEL. Betätigen Sie MENU um die Menümarken zu entfernen und erhalten Sie eine volle Ansicht vom Diagramm. Betätigen Sie NXT NXT PICT CANCL um zurückzukehren zum PLOT WINDOW-FUNCTION-Funktion. Betätigen Sie ENTER um zur normalen Rechneranzeige zurückzugehen.

Variable PPAR

Betätigen Sie VAR um Ihr Variablen Menü bei Bedarf zurückzugewinnen. In Ihrem Variablen Menü sollten Sie eine Variable haben, die als PPAR markiert wird. Betätigen Sie ↱ PPAR um der Inhalt dieses Variablen im Stapel zu erhalten. Betätigen Sie den Untenpfeil-Tasten, um den Stapelherausgeber zu starten, und verwenden Sie die hohen und Untenpfeil-Tasten, um den vollen Inhalt von PPAR anzusehen. Der Schirm zeigt die folgenden Werte:

PPAR steht für Plot-Parameter, und sein Inhalt umfaßt zwei bestellte Paare der realen Zahlen, (-8, -1,10797263281) und (2., 7,38905609893),

welches *beziehungsweise die Koordinaten der Ecke der untereren Linke und der oberen rechten Ecke* des Plots darstellt. Zunächst verzeichnet PPAR den *Namen der unabhängigen Variable*, X, gefolgt von einer Zahl, die die *Stufensprung der unabhängigen Variable* im Erzeugung des Plots spezifiziert. Der Wert, der hier gezeigt wird, ist der Default-Wert, null (0.), das Stufensprünge in X spezifiziert, 1 Pixel in der Graphikanzeige entsprechend. Das nächste Element in PPAR ist *eine Liste die zuerst die Koordinate von dem Koinzidenzpunkt enthält vom Durchschnitt der Plotäxte*,d.h., (0., 0.), *gefolgt durch eine Liste die Anmerkungen von Tick-Markierungen* spezifiziert auf beziehungsweise dem x und den Y-axes respectively {# 10d # 10d}. Zunächst verzeichnet PPAR *die Sorte des Plots*, die erzeugt werden muss d.h. FUNCTION, und schließlich die *Y-axismarkierung* d.h. Y.

Das variable PPAR, wenn nicht vorhanden, wird erzeugt, jedesmal wenn Sie einen Plot machen. Der Inhalt der Funktion ändert, abhängig von der Sorte des Plots und auf den Optionen, die Sie auserwählten im PLOT-Fenster (das

Fenster erzeugt durch die simultane Aktivierung von ⬅ und WIN (F2) Tasten.

Umgekehrte Funktionen und ihre Diagramme

Lassen Sie *y = f(x)*, wenn wir eine Funktion *y = g(x)* finden können so daß, *g(f(x)) = x*,dann sagen wir, daß *g(x)* die *umgekehrte Funktion* von *f(x)*.ist. Gewöhnlich wird die Darstellung $g(x) = f^{-1}(x)$ benutzt, um eine umgekehrte Funktion zu bezeichnen. Mit dieser Darstellung können wir schreiben : wenn *y = f(x),* dann $x = f^{-1}(y)$. Auch, $f(f^{-1}(x)) = x$, and $f^{-1}(f(x)) = x$.

Wie früher angezeigt, die Funktionen ln(x) und exp(x) sind Gegenteile von einander, d.h., ln(exp(x)) = x, und exp(ln(x)) = x. Dieses kann überprüft werden in dem Rechner durch folgende Ausdrucken in dem Gleichungsverfasser zu schreiben und auszuwerten. LN(EXP(X)) und EXP(LN(X)). Die müssten beide zu X auswerten.

Wenn eine Funktion *f(x)* und seine umgekehrte $f^{-1}(x)$ gleichzeitig im gleichen Satz der Äxte geplottet werden, sind ihre Diagramme Reflexionen von einander über die Linie line *y = x*. Lassen wir uns diese Tatsache mit dem Rechner auf die Funktionen LN(X) und EXP(X) durch dieses Verfahren überprüfen.

Betätigen Sie, gleichzeitig wenn in RPN Modus, ⬅ Y=. Die Funktion function Y1(X) = EXP(X) sollt vorhanden sein in dem PLOT-FUNCTION Fenster von die vorhergehende Übung. _ Betätigen Sie ADD und tippen Sie die Funktion `Y2(X) = LN(X)`. Auch laden Sie die Funktion Y3(X) = X. Betätigen Sie NXT OK um zum normalen Rechneranzeige zurückzugehen.

Betätigen Sie, gleichzeitig wenn in RPN Modus, ⬅ WIN , und änderen Sie H-View-Strecke um lesen zu können: H-VIEW: -8 8

Betätigens Sie AUTO um die vertikale Strecke zu erzeugen. Betätigen Sie ERASE DRAW um das Diagramm von y = ln(x), y = exp(x), und y =x zu produzieren, gleichzeitig wie im RPN Modus.

Sie beachten, daß nur das Diagramm von y = exp(x) offenbar sichtbar ist. Etwas paßte nicht zum AUTO Vorwähler der vertikalen Strecke. Was geschieht ist, wenn Sie AUTO in der PLOT-FUNKTION-WINDOW-Schirm betätigen, produziert der Rechner die vertikale Strecke, die der ersten Funktion entspricht in der Liste der geplottet werden muss. Welches in diesem Fall geschieht, Y1(X) = EXP(X) zu sein. Wir müssen die vertikale Strecke selbst eintragen, um die anderen zwei Funktionen im gleichen Plot anzuzeigen.

Betätigen Sie CANCL um in dem PLOT-FUNCTION-WINDOW-Schirm zurückzukehren. Ändern Sie die vertikalen und horizontalen Strecken, um zu lesen: H-VIEW: -8 8, V-VIEW: -4 4

Indem wir diese Strecken vorwählen, stellen wir sicher, daß der Skala des Diagramms 1 Vertikal bis 1 Horizontal gehalten is.t Betätigen Sie ERASE DRAW und Sie erhalten die Plots der natürlichen exponentialen und y = x Funktionen des Logarithmus. Es ist vom Diagramm offensichtlich klar, daß LN(X) und EXP(X) Reflexionen von einander sind über die Linie y = X. Betätigen Sie CANCL um zum PLOT-WINDOW-FUNCTION zurückzugehen. · Betätigen Sie ENTER um zur normalen Rechneranzeige zurückzugehen.

Zusammenfassung des FUNCTION Plotarbeiten

In diesem Abschnitt stellen wir die Informationen dar betreffend PLOT SETUP, PLOT-FUNCTION, und die PLOT WINDOW-Schirme, die zugänglich sind durch linksverschiebenden Taste, der mit den Weichmenü Tasten F1 bis F4 kombiniert wird. Gegründet auf den oben graphisch dargestellten Beispielen, das Verfahren, um zu folgen, einen FUNCTION Plot zu produzieren (d.h. einer der eine oder mehr Funktionen der Form Y = F(X) plottet), , ist das folgende

← 2D/3D (gleichzeitig, wenn im RPN Modus). Zugang zum PLOT SETUP-Fenster Wenn sie benötigt werden, ändern Sie `TYPE` zur `FUNCTION`, und tragen Sie den Namen der unabhängigen Variable ein.

Einstellungen:

- Überprüfung auf `_Simult` bedeutet daß Simult bedeutet daß, wenn Sie zwei oder mehr Plots im gleichen Diagramm haben, sie wird geplottet gleichzeitig, wenn sie das Diagramm produziert.

- Überprüfung auf _ `Connect` bedeutet daß die Kurve eine ununterbrochene Kurve anstatt ein Satz einzelne Punkte ist.
- Überprüfung auf `_Pixels` bedeutet, daß die Markierungen, die durch angezeigt werden, werden getrennt durch die viele Pixel H-Ticken und V-Ticken.
- Der Default-Wert für beide vorbei `H-Tick` und `V-Tick` ist 10. Frei belegbare Funktionstaste Menüwahlen:

Frei belegbare Funktionstaste Menüwahlen:

- Benützen Sie EDIT um Aufbereitungsfunktionen von Werten auf dem vorgewählten Gebiet zu redigieren.
- Benützen Sie CHOOS, die Art des Plots vorzuwählen zu verwenden, wenn die `Type`: Feld wird hervorgehoben. Für die gegenwärtigen Übungen wünschen wir dieses Feld einzustellen, auf Funktion.

> **Anmerkung**: die weichen MenüTasten EDIT und CHOOS sind nicht gleichzeitig vorhanden. Ein oder das andere werden vorgewählt, abhängig vom dem Feld, einzugeben hervorgehoben wird.

- Betätigen Sie der weiche Menütasten der AXES zum Vorwählen oder das Plotten der Äxte im Diagramm abzuwählen. Wenn die Option 'Plotter Äxte' vorgewählt wird, erscheint ein quadratischer Punkt im Tastenmarkierung: AXES■ Das Fehlen der quadratischen Punkt zeigt an, daß Äxte nicht im Diagramm geplottet werden.
- Betätigen Sie ERASE um irgend ein Diagramm zu löschen, daß im Graphikanzeige Fenster besteht.
- Betätigen Sie DRAW um das Diagramm entsprechend dem gegenwärtigen Inhalt von PPAR für die verzeichnete Gleichungen zu produzieren im PLOT-FUNCTION Fenster.
- Betätigen Sie NXT, zum des zweiten Satzes der weichen Menütasten auf diesem Schirm zugänglich zu machen.
- Btätigen Sie RESET, um irgend ein vorgewähltes Feld zu seinem Default-Wert zurückzustellen.
- Betätigen Sie CANCL, um alle mögliche Änderungen am PLOT SETUP Fenster zu annullieren und geht zur normalen Rechneranzeige zurück.
- Betätigen Sie OK zum der Änderungen an den Wahlen im PLOT zu speichern SETUP Fenster und kommt zur normalen Rechneranzeige zurück.

⬅ Y= (gleichzeitig, wenn im RPN Modus). . Machen Sie das PLOT-Fenster zugänglich (in diesem Fall wird es PLOT - FUNCTION Fenster genannt).

Frei belegbare Funktionstaste Menüwahlen:

- Betätigen Sie EDIT um die hervorgehobene Gleichung zu redigieren.
- Betätigen Sie ADD um neue Gleichungen dem Plot beizufügen.

> **Anmerkung**: ADD oder EDIT löst den Gleichungsverfasser EQW aus, den Sie verwenden können, um neue Gleichungen zu schreiben oder alte Gleichungen zu redigieren.

- Betätigen Sie DEL um die hervorgehobene Gleichung zu entfernen.
- Betätigen Sie CHOOS um eine Gleichung zu addieren, die bereits in Ihrem Variablen Menü definiert wird, aber verzeichnet nicht im PLOT-FUNCTION Fenster.
- Betätigen Sie ERASE um irgend ein Diagramm zu löschen, daß im Graphikanzeige Fenster besteht.
- Betätigen Sie DRAW um das Diagramm entsprechend dem gegenwärtigen Inhalt von PPAR für die verzeichnete Gleichungen zu produzieren im PLOT-FUNCTION Fenster.
- Presse NXT um der zweiten Menüliste zu aktivieren.
- Betätigen Sie MOVE↑ und MOVE↓ um die vorgewählte Gleichung beziehungsweise eine Position oben oder nach unten zu bewegen,.
- Betätigen Sie CLEAR, wenn Sie zum freien Raum alle Gleichungen wünschen z.Z., die im PLOT - FUNCTION Fenster aktiv sind. Der Rechner überprüft, ob oder nicht Sie zum freien Raum alle Funktionen wünschen, bevor Sie alle löschen. Wählen Sie YES vor und betätigen Sie OK um dem Löschen aller Funktionen fortzufahren. Wählen Sie NO vor und betätigen Sie OK um die Option CLEAR zu deactivieren..
- Betätigen Sie OK wenn Sie fertig sind, um zur normalen Rechneranzeige zurückzugehen.

⬅ WIN (gleichzeitig, wenn im RPN Modus). Zugang zum PLOT WINDOW-Fenster.

Einstellungen:

- tragen Sie die untereren und oberen Begrenzungen für horizontale Ansicht (H-VIEW) und vertikale Strecken der Ansicht (V-VIEW) in das Plotfenster ein. Oder,
- tragen Siedie untereren und oberen Begrenzungen für horizontale Ansicht (H-VIEW) ein und betätigen AUTO,, während der Cursor in einem der V-VIEW Felder ist, um die vertikale Strecke der Ansicht (V-VIEW) automatisch zu erzeugen. Oder,
- tragen Sie die untereren und oberen Begrenzungen für vertikale Ansicht (V-VIEW) ein und betätigen AUTO,, während der Cursor in einem der H-VIEW Felder ist, um die horizontale Strecke der Ansicht (H-VIEW) automatisch zu erzeugen.
- der Rechner benutzt die horizontale Strecke der Ansicht (H-VIEW), um Datenwerte für das Diagramm zu erzeugen, es sei denn daß Sie die Optionen Indep Tief ändern, (Indep), Höhe und (Indep) Schritt ändern. Diese Werte stellen, beziehungsweise, das Minimum, das Maximum und die Stufensprungwerte der unabhängigen fest im Plot verwendet zu werden Variable. Wenn die Option Rückstellung auf den Gebieten Indep niedrig verzeichnet wird, (Indep) stark und (Indep) Schritt, verwendet der Rechner das Minimum und die Maximalwerte, die von H-View festgestellt werden.

- Überprüfung auf _Pixels bedeutet, daß die Werte der unabhängigen variablen Stufensprünge (Schritt:) werden in den Pixeln anstatt in den Plotkoordinaten gegeben.

Frei belegbare Funktionstaste Menüwahlen:

- Betätigen Sie EDIT um irgendeine Eintragung im Fenster zu redigieren.
- Betätigen Sie AUTO wie oben in den Einstellungen erklärt.
- Betätigen Sie ERASE um irgend ein Diagramm zu löschen, daß im Graphikanzeige Fenster besteht.
- Betätigen Sie DRAW um das Diagramm entsprechend dem gegenwärtigen Inhalt von PPAR für die verzeichnete Gleichungen zu produzieren im PLOT-FUNCTION Fenster.

- Presse NXT um der zweiten Menüliste zu aktivieren.
- Betätigen Sie RESET um irgendein vorgewähltes Feld zu seinem Default-Wert zurückzustellen.
- Benutzen Sie CALC um der Rechnerstapel zugänglich zu machen, um Berechnungen durchzuführen, die notwendig sein können, um einen Wert für eine der Optionen in diesem Fenster zu erhalten. Wenn der Rechnerstapel zugänglich gemacht ist, haben Sie auch die frei belegbare Funktionstasten CANCL und OK .
- Benutzen Sie CANCL wann Sie die derzeitigen Berechnung annullieren wollen und zurück gehen zur PLOT WINDOW-Schirm. Oder,
- Benutzen Sie OK um die Resultaten von Ihrer Berechnung zu akzeptieren und zurürck zu gehen zur PLOT WINDOW-Schirm.
- Benutzen Sie TYPES um Informationen über die Art der Gegenstände zu erhalten, die auf dem vorgewählten Optionengebiet benutzt werden können.
- Betätigen Sie CANCL, um alle mögliche Änderungen am PLOT WINDOW-Schirm zu annullieren und geht zur normalen Rechneranzeige zurück.
- Betätigen Sie OK um alle mögliche Änderungen am PLOT WINDOW-Schirm zu akzeptieren und geht zur normalen Rechneranzeige zurück.

← GRAPH (gleichzeitig, wenn im RPN Modus). Plottet das Diagramm, das auf den Einstellungen basiert, die in variablem PPAR und gegenwärtigen in den Funktionen definiert werden im PLOT-FUNCTION Schirm gespeichert werden. Wenn ein Diagramm, unterschiedlich zu dem, das Sie plotten, bereits auf dem graphischen Anzeigeschirm, besteht der neue Plot wird gelegt auf dem vorhandenen Plot. Dieses kann nicht sein Resultat Sie wünscht folglich empfehle mich ich, die ERASE DRAW zu verwenden weichen MenüTaste, die in den PLOT-SETUP- vorhanden sind, PLOT-FUNCTION- oder PLOT WINDOW-Schirmen.

Plots der trigonometrischen und hyperbelfunktionen

Die Verfahren, die oben verwendet werden, um LN(X) und EXP(X) zu plotten, separat oder gleichzeitig, können verwendet werden, um jede mögliche Funktion der Form zu plotten y = f(x). Sie wird wie eine Übung zum Leser gelassen, um die Plots der trigonometrischen und Hyperbelfunktionen und ihrer Gegenteile zu produzieren. Die Tabelle unten folgend, schlägt die

Werte vor, um für die vertikalen und horizontalen Strecken in jedem Fall zu verwenden. Sie können die Funktion Y=X umfassen, wenn Sie gleichzeitig eine Funktion und sein Gegenteil plotten, um ihre ' Reflexion ' über die Linie zu überprüfen Y = X.

	H-VIEW Strecke:		V-VIEW Strecke:	
Funktion:	**Minimum**	**Maximum a**	**Minimum**	**Maximum a**
SIN(X)	-3.15	3.15	AUTO	
ASIN(X)	-1.2	1.2	AUTO	
SIN & ASIN	-3.2	3.2	-1.6	1.6
COS(X)	-3.15	3.15	AUTO	
ACOS(X)	-1.2	1.2	AUTO	
COS & ACOS	-3.2	3.2	-1.6	1.6
TAN(X)	-3.15	3.15	-10	10
ATAN(X)	-10	10	-1.8	1.8
TAN & ATAN	-2	-2	-2	-2
SINH(X)	-2	2	AUTO	
ASINH(X)	-5	5	AUTO	
SINH & ASINH	-5	5	-5	5
COSH(X)	-2	2	AUTO	
ACOSH(X)	-1	5	AUTO	
COS & ACOS	-5	5	-1	5
TANH(X)	-5	5	AUTO	
ATANH(X)	-1.2	1.2	AUTO	
TAN & ATAN	-5	5	-2.5	2.5

Eine Tabelle der Werte für eine Funktion erzeugen

Die Kombinationen ← TBLSET (F5) und ← TABLE (F6), gleichzeitig betätigd wenn in RPN Modus, laßt der Benutzer eine Tabelle der Werte für Funktionen erzeugen. Z.B. wir werden eine Tabelle produzieren der Funktion Y(X) = X/(X+10), in der Strecke -5 < X < 5 diesen Anweisungen folgend:

- Wir werden Werte erzeugen von Funktion f(x), wie hier oben definiert, mit Werten für x von –5 bis 5, in Stufensprünge von 0,5. Zuerst müssen wir sichergehen, daß die Diagrammtype eingestellt wird auf **FUNCTION**, im PLOT-SETUP-Schirm (2D/3D, betätigen sie gleichzeitig, wenn im RPN Modus). Das Feld vor der *Type* Option wird hervorgehoben. Wenn dieses Feld noch nicht eingestellt ist auf **FUNCTION** betätigen Sie die freibelegte Taste CHOOS und wählen Sie die **FUNCTION** Option, dann betätigen Sie OK.
- Zunächst betätigen Sie ▼ um das Feld vor der Wahl EQ hervorzuheben, und den Funktion Ausdruck zu schreiben: 'X',
- Um die Änderungen im PLOT SETUP Schirm zu akzeptieren, betätigen Sie NXT OK. Sie werden zurück kommen im normalen Rechneranzeiger.
- Der folgende Schritt ist den Tabelle-Einstellungsschirm zugänglich zu machen, indem er den Tastenanschlag combination TBLSET gleichzeitig verwendet (d.h., frei belegbare Funktionstaste F5) - wenn im RPN Modus. Dieses produziert einen Schirm, in dem Sie den Anfangswert (Anfang) und die Stufensprung (Schritt) vorwählen können. Tragen Sie folgendes ein: 5 +/- OK 0 • 5 OK 0 • 5 OK (d.h. der Zoomfaktor = 0,5). Schalten Sie die freibelegter ✓CHK Menütaste um, bis eine überprüfende Markierung vor dem Option *kleinen Schriftkegel* erscheint, wenn Sie daßso wünschen. Betätigen Sie OK. Sie werden zurück kommen im normalen Rechneranzeiger.

Das Variable TPAR

Nach der Fertigung der Tabelle, die aufgestellt wird, verursacht Ihr Rechner eine Variable, die TPAR angerufen wird (Tabelle Parameter) die Informationen speichert die relevant zum Tabelle, das erzeugt werden soll. Um den Inhalt dieser Variable zu sehen, betätigen Sie TPAR.

- Um die Tabelle zu sehen, betätigen Sie TABLE (d.h., freibelegter Menütaste F6) – gleichzeitig wenn in RPN Modus.menu Taste Dieses produziert eine Tabelle von Werten von *x = -5, -4,5*... und die entsprechenden Werte von f(x), verzeichnet als Y1 durch Rückstellung. Sie können die Pfeiltasten auf und ab verwenden, um ungefähr in die

Tabelle umzuziehen. Sie beachten, daß wir nicht einen Endewert für die unabhängige Variable x anzeigen mußten. So fährt die Tabelle über dem Maximalwert für x hinaus fort, daß früh vogeschlagen wurde, nämlich x = 5.

Manche Optionen Die zugänglish sind, weil die Tabelle sichtbar ist, sind ZOOM, BIG,und DEFN:

- das DEFN, wenn es vorgewählt wird, zeigt die Definition der unabhängigen Variable.
- der BIG Taste ändert einfach den Schriftkegel in der Tabelle von kleinem zu grossem und umgekehrt. Versuchen Sie es mal.
- der ZOOM Taste, wenn Sie betätigt werden, produziert ein Menü mit den Wahlen: *In, heraus, Dezimalstrich, Ganzzahl und Triglyzerid* Versuchen Sie folgende Übungen:

    - Mit der Wahl, *In* hervorgehoben, betätigen Sie OK. Die Tabelle wird erweitert, damit die X-Stufensprung jetzt 0,25 anstatt 0,5 ist. Einfach was der Rechner macht, ist die ursprüngliche Stufensprung, 0,5, mit dem Zoomfaktor, 0,5 multiplizieren, um die neue Stufensprung von 0,25 zu produzieren. So ist der *zoom in* Option nützlich, wenn Sie mehr Auflösung für die Werte von x in Ihrer Tabelle wünschen.
    - Um der Auflösung durch einen zusätzlichen Faktor von 0,5 zu erhöhen, betätigen Sie ZOOM, noch einmal *In* auserwählen und betätigen Sie OK.. Die X - Stufensprung ist jetzt 0,0125.
    - Um der vorhergehenden X-Stufensprunges zurückzugewinnen, betätigen Sie ZOOM ▲ OK, um die Wahl *Un-zoom* vorzuwählen. Die X - stufensprung ist jetzt vergrößert zu 0,25.
    - Um der ursprünglichen X-Stufensprunges von 0,5 zurückzugewinnen, was Sie tun können ist *un-zoom* wieder oder benutzen Sie den *option zoom out* durch betätigendes ZOOM OK.
    - Die Option Dezimalstrich in ZOOM produziert X-Stufensprünge von 0,10.
    - Die Option Dezimalstrich in ZOOM produziert X-Stufensprünge von 1.

- Die Option Trig in produziert Stufensprünge bezogen auf Brüchen von π, die nützlich, wenn trigonometrische Funktionen geplottet werden.
- Um zurück zu gehen zur normalen Rechneranzeige, betätigen Sie ENTER .

Plots in den polaren Koordinaten

Zuerst von allen, können Sie die Variablen gelöscht wünschen, die in den vorhergehenden Beispielen verwendet wurden (z.B., X, EQ, Y1, PPAR) dieFunktion PURGE verwendend (TOOL PURGE). Indem Sie dies tun, werden alle Parameter, die auf Graphiken bezogen werden, gelöscht. Betätigen Sie VAR um zu prüfen, ob die Variablen in der Tat bereinigt wurden.

Wir versuchen, die Funktion f(θ) = 2(1-sin(θ)), wie folgt zu plotten:

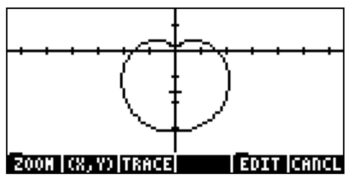
- Zuerst überprüfen Sie, ob Ihr Rechner's Winkelmaß auf Einheitswinkel eingestellt ist.
- Betätigen Sie ← 2D/3D gleichzeitig, wenn im RPN Modus, um das PLOT SETUP-Fenster zu erhalten.
- Ändere `TYPE` in `Polar` CHOOS ▼ OK. betätigend.
- Betätigen Sie ▼ und tippen Sie:

  ' 2 × ← () 1 — SIN ALPHA → T OK.

- Der Cursor ist jetzt im `Indep` Feld. Betätigen Sie ' ALPHA → T OK um das unabhängige Variable zu ändern in θ.
- Betätigen Sie NXT OK um zur normalen Rechneranzeige zurückzugehen.
- Betätigen Sie ← WIN gleichzeitig wenn in RPN Modus, um das PLOT-Fenster zu erhalten (in diesem Fall wird er genannt PLOT-POLAR Fenster).
- Änderen Sie die H-View-Strecke von -8 bis 8, durch das Verwenden von 8 +/- OK 8 OK, und und die V-VIEW-Strecke von -6 bis 2 durch das Verwenden von 6 +/- OK 2 OK.
-

Anmerkung: die H-VIEW und die V-VIEW stellen die Skalen nur des Anzeige- Fensters fest, und ihre Stecken hängen in diesem Fall nicht mit der Strecken der Werte der unabhängigen Variablen zusammen.

- Änderen Sie der `Indep Low` Wert bis 0 und der hohe Wert bis 6,28 (≈ 2π), durch das Verwenden von: durch das Verwenden von 0 +/- OK 6 . 2 8 OK.
- Betätigen Sie ERASE DRAW um der Funktion in den polaren Koordinaten zu plotten. Das Resultat ist eine Kurve, die wie ein Herz geformt ist. Diese Kurve bekannt als *cardiod* (`cardios`, Griechisch für Herz).
-

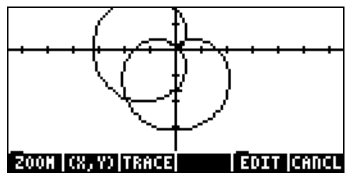


- Betätigen Sie EDIT NXT LABEL MENU um das Diagramm mit Markierung zu sehen. Betätigen Sie NXT um zurückzugehen im Menü. Betätigen Sie NXT PICT um das ursprüngliche graphische Menü zurückzugewinnen.
- Betätigen Sie TRACE (X,Y) um der Kurve zu verfolgen. Die Daten, die an der Unterseite der Anzeige gezeigt werden, sind der Winkel θ und der Radius r, obgleich das letzte Y markiert wird (Rückstellung Namen der abhängigen Variable).
- Betätigen Sie NXT CANCL um in dem PLOT WINDOW-Schirm zurückzukehren. Betätigen Sie NXT OK um zur normalen Rechneranzeige zurückzugehen.

In dieser Übung trugen wir die direkt im PLOT-SETUP-Fenster geplottet zu werden Gleichung. Wir können Gleichungen für das Plotten mit dem PLOT-Fenster auch eintragen d.h. gleichzeitig wenn im RPN Modus und ← Y= betätigen. Nach der Fertigung der vorhergehenden Übung, z.B. wenn Sie ← Y= betätigen, erhalten Sie Gleichung '2*(1-SIN(θ))' hervorgehoben. Sagen wir, wir möchten die Funktion '2*(1-COS(θ))' zusammen mit der vorhergehenden Gleichung auch plotten.

- Betätigen Sie ADD und tippen Sie 2 × ← () 1 − COS ALPHA → T ENTER, um die neue Gleichung einzutragen.

- Betätigen Sie ERASE DRAW um die zwei Gleichungen in der gleichen Abbildung geplottet zu sehen. . Das Resultat ist zwei schneidende *cardioids.* Betätigen Sie CANCL ON um zur normalen Rechneranzeige zurückzugehen.
-



Plotten der konischen Kurven

Die allgemeinste Form einer konischen Kurve in der x-y Fläche ist: Wir erkennen auch als konische Gleichungen die, die in der kanonischen Form für die folgenden Abbildungen gegeben werden:

- Kreis: $(x-x_o)^2+(y-y_o)^2 = r^2$
- Ellipse: $(x-x_o)^2+(y-y_o)^2 = 1$
- Parabel: $(y-b)^2 = K(x-a)$ or $(x-a)^2 = K(y-b)$
- Hyperbel: $(x-x_o)^2+(y-y_o)^2 = 1$ oder $xy = K$,

wo x_o, y_o, *a, b*, und *K* konstant sind.

Die konischen Namenskurven folgt, weil diese Abbildungen (Kreise, Ellipses, Parabeln oder Hyperbeln) aus dem Durchschnitt einer Fläche mit einem Kegel resultieren. Z.B. ist ein Kreis der Durchschnitt eines Kegels mit einem Flächesenkrechten zur Hauptmittellinie des Kegels.

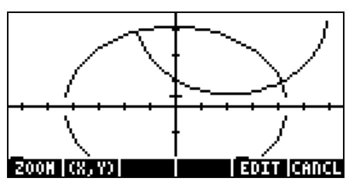
Der Rechner läßt die Fähigkeit von einer oder mehr plottender konischen Kurven, indem er konisch als die Funktion TYPE im PLOT-Klima vorwählt. Überprüfen Sie, die Variablen PPAR und EQ zu löschen, bevor Sie fortfahren. Z.B. lassen Sie uns die Liste von Gleichungen speichern

{ ‘(X-1)^2+(Y-2)^2=3’ , ‘X^2/4+Y^2/3=1’ }

in das Variable EQ.

Diese Gleichungen, die wir eines Kreises erkennen als die, der zentriert wird an (1.2) mit Radius √3 und eines Ellipse zentriert an (0.0) mit Halbmittellinie Längen a = 2 und b = √3.

- Betreten Sie das PLOT-Klima, durch betätigendes [←] 2D/3D gleichzeitig wenn im RPN Modus, und wählen Sie `Conic` als den `TYPE` vor. . Die Liste von Gleichungen wird auf dem EQ Gebiet verzeichnet.
- Überprüfen Sie, ob die unabhängige Variable (`Indep`) auf ' X ' und die abhängige Variable (`Depnd`) auf ' Y ' eingestellt werden.
- Betätigen Sie [NXT] OK um zur normalen Rechneranzeige zurückzugehen.
- Tragen Sie das PLOT WINDOW-Klima ein, indem Sie gleichzeitig [←] WIN betätigen wenn Sie im RPN Modus sind.
- Änderen Sie die Strecke für H-VIEW bis -3 bis 3, durch das Verwenden von [3] [+/-] OK [3] OK. Änderen Sie auch die Strecke für H-VIEW bis -1,5 bis 2, durch das Verwenden von [1] [•] [5] [+/-] OK [2] OK.
- Änderen Sie *Indep Low* und *High:* Felder zum Zurückfallen durch das Verwenden von [NXT] RESET, während jedes jener Felder hervorgehoben wird. Wählen Sie die Option vor *Reset value* nach betätigendem RESET. Betätigen Sie OK um das Zurückstellen von Werten durchzuführen. Betätigen Sie [NXT] um zum Hauptmenü zurückzugehen.
- Plotten Sie das Diagramm: ERASE DRAW .
-

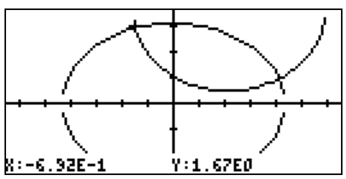


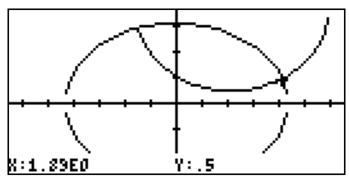
Anmerkung: Die `H-View` und `V-View` Strecken wurden vorgewählt, um den Durchschnitt der zwei Kurven zu zeigen. . Es gibt keine allgemeine Richtlinie zum vorwählen jener Strecken, ausgenommen gegründet auf was wir von die Kurven kennen. Z.B. für die Gleichungen, die oben gezeigt werden, wissen wir, daß der Kreis von -3+1 = von -2 bis von 3+1 = von 4 in x verlängert, und von -3+2=-1 zu 3+2=5 in y. Zusätzlichverlängert der

Ellipse, der am Ursprung (0,0) zentriert wird, von -2 bis 2 in x, und von -√3 zu √3 in y.

Beachten Sie, daß für den Kreis und den Ellipse die Region, die den linken und rechten Übermaßen der Kurven entspricht, nicht geplotted werden. Dieses ist der Fall mit allen Kreisen oder Ellipses geplotteten Verwenden, die als `TYPE Conic` sind.

- Um Markierungen zu sehen: EDIT NXT LABEL MENU
- Um das Menü zurückzugewinnen. NXT NXT PICT
- Um der Koordinaten des Koinzidenzpunkts zu schätzen, betätigen Sie die (X,Y) Menütaste und verschieben den Cursor so nahe, wie möglich auf jene Punkte mit den Pfeiltasten. Die Koordinaten des Cursors werden in der Anzeige gezeigt. Z.B. ist der linke Koinzidenzpunkt nah an (-0,692, 1,67), während der rechte Durchschnitt nahe ist (1,89.0,5).





- Betätigen Sie NXT CANCL um das Menü zurückzugewinnen und zum PLOT-Klima zurückzugehen.
- Betätigen Sie NXT OK um zur normalen Rechneranzeige zurückzugehen.

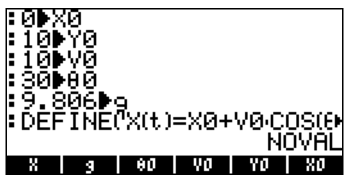
Parametrische Plots

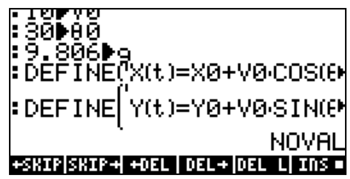
Parametrische Plots in der Fläche sind jene Plots deren Koordinaten durch das System von Gleichungen x = x(t) und y = y(t) erzeugt werden, in dem t als der Parameter bekannt ist. . Ein Beispiel solchen Diagramms ist die Flugbahn eines Geschosses, $x(t) = x_0 + v_0 \cdot COS\ \theta_0 \cdot t$, $y(t) = y_0 + v_0 \cdot \sin\ \theta_0 \cdot t - \frac{1}{2} \cdot g \cdot t^2$. Gleichungen plotten mögen Sie diese, die Konstante x_0, y_0, v_0, and θ_0, miteinbeziehen, wir müssen die Werte jener Parameter in den Variablen speichern. Um dieses Beispiel zu entwickeln, stellen Sie ein Unterverzeichnis her, das ' PROJM ' für Wurfbewegung und innerhalb dieses Unterverzeichnisspeichers die folgenden Variablen genannt wird: X0 = 0, Y0

= 10, V0 = 10 , θ0 = 30, and g = 9.806. Überprüfen Sie, ob das Winkelmaß des Rechners auf DEG eingestellt wird. . Zunächst definieren Sie Funktionen (use ← DEF):

$$X(t) = X0 + V0*COS(\theta 0)*t$$
$$Y(t) = Y0 + V0*SIN(\theta 0)*t - 0.5*g*t\char`^2$$

welches das Variablen Y und das X den weichen MenüTastemarken hinzufügt.



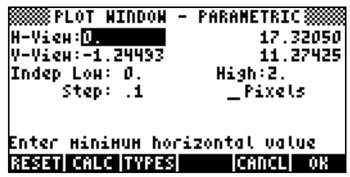


Um das Diagramm selbst zu produzieren, folgen Sie diesen Schritten:

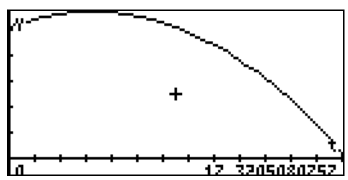
- Zunächst betätigen Sie ← 2D/3D gleichzeitig, wenn im RPN Modus, um das PLOT-SETUP-Fenster zu sehen .
- Änderen Sie `TYPE` zu `Parametric`, durch CHOOS ▼ ▼ OK.
- Betätigen Sie ▼ und tippen Sie 'X(t) + i*Y(t)' OK um der parametrischen Plot als der einer komplexen Größe zu definieren. (die realen und Imaginärteile der komplexen Größe entsprechen dem x und y-koordinieren von der Kurve.)
- Der Cursor ist jetzt auf dem `Indep` Gebiet. Betätigen Sie ' ALPHA → T OK um das unabhängige Variable zu ändern in *t.*
- Betätigen Sie NXT OK um zur normalen Rechneranzeige zurückzugehen.
- Betätigen Sie ← WIN gleichzeitig wenn in RPN Modus, um das PLOT-Fenster zu erhalten (in diesem Fall wird er genannt PLOT-PARAMETRIC Fenster). Anstatt, das horizontale und die Vertikale zu ändern an, sieht zuerst, wie für andere Sorten Plot getan, wir die untereren und oberen Werte der unabhängigen Variable zuerst einstellen, wie folgt:
- wählen das `Indep Low` Feld vor, indem Sie betätigen mit ▼▼ Änderen Sie dieser Wert in 0 OK. Dann ändern Sie den Wert von `High` zu to 2 OK. Geben Sie ein. 0 . 1 OK für den Schrittwert (d.h., Schritt = 0,1).

Anmerkung: Durch diese Einstellungen zeigen wir, daß der Parameter t Werte von t = 0, 0.1, 0.2 nimmt…, etc., bis das Erreichen von 2.0 an.

- Betätigen Sie AUTO. Dieses erzeugt automatische Werte der H-VIEW und V-VIEW Strecken, die auf den Werten der unabhängigen Variable t und die Definitionen von verwendetem X(t) und Y(t) basieren. Das Resultat ist:



- Betätigen Sie ERASE DRAW um die parametrischen Plot zu zeichnen.
- Betätigen Sie EDIT NXT LABEL MENU um das Diagramm mit Markierung zu sehen. Die Fensterparameter sind so, daß Sie nur Hälfte der Label im X-axis sehen.



- Betätigen Sie NXT um zurückzugehen im Menü. Betätigen Sie NXT PICT um das ursprüngliche graphische Menü zurückzugewinnen.
- Betätigen Sie TRACE (X,Y) um Koordinaten irgendeines Punktes auf dem Diagramm festzustellen. Benutzen Sie ▶ und ◀, um den Cursor über die Kurve zu verschieben. Am unteren Bildschirmrand sehen Sie den Wert des Parameters t und Koordinaten des Cursors wie (X,Y).
- Betätigen Sie NXT CANCL um in dem PLOT WINDOW-Schirm zurückzukehren. Dann betätigen Sie ON oder NXT OK um zur normalen Rechneranzeige zurückzugehen.

Ein Bericht Ihrer freigehobenen MenüTastetasten zeigt, daß Sie jetzt die folgenden Variablen haben: t, EQ, PPAR, Y, X, g, θ0, V0, Y0, X0. Variablen t, EQ und PPAR werden durch den Rechner erzeugt, um die gegenwärtigen

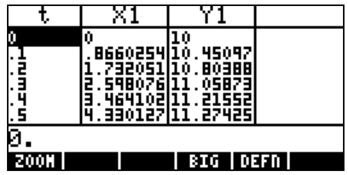
Werte des Parameters, des t, der Gleichung zu speichern, um geplottete EQ (die ' X(t enthält), + I*Y(t) ') und die Plotparameter zu sein. . Die anderen Variablen enthalten die Werte der Konstanten, die in den Definitionen von X(t) und von Y(t) verwendet werden.

Sie können unterschiedliche Werte in den Variablen speichern und neue parametrische Plots der Wurfgleichungen produzieren, die in diesem Beispiel verwendet werden. Wenn Sie den gegenwärtigen Abbildung Inhalt löschen möchten, bevor Sie einen neuen Plot produzieren, müssen Sie entweder den PLOT zugänglich machen, PLOT WINDOW, oder, PLOT SETUP-Schirme sortiert, durch das Betätigen, ← Y= , ← WIN , oder ← 2D/3D aus (die zwei Taste müssen wenn im RPN Modus gleichzeitig betätigt werden). Dann betätigens Sie ERASE DRAW . Betätigen Sie CANCL, um zum PLOT, PLOT WINDOW oder PLOT SETUP-Schirm zurückzugehen . Dann betätigen Sie ON, oder NXT OK, um zur normalen Rechneranzeige zurückzugehen.

Eine Tabelle erzeugen für parametriche Gleichungen

In einem früheren Beispiel erzeugten wir eine Tabelle der Werte (X,Y) für einen Ausdruck der Form Y=f(X) d.h. eine Funktion Sorte von Diagramm. In diesem Abschnitt stellen wir das Verfahren für das Erzeugen einer Tabelle dar, die einem parametrischen Plot entspricht. Zu diesem Zweck ziehen wir Nutzen aus den parametrischen Gleichungen, die oben im Beispiel definiert werden.

- Zuerst, lassen wir das TABLE SETUP-Fenster durch gleichzeitig betätigendes ← TBLSET zugänglich machen, wenn im RPN Modus. Für die unabhängige variable Ändern Sie der Anfangswert auf 0,0 und den Schrittwert zu 0,1, Betätigen Sie OK.
- erzeugen die Tabelle, indem es, gleichzeitig wenn im RPN Modus, ← TABLE betätigt. Die resultierende Tabelle hat drei Spalten, den Parameter t darzustellen, und die Koordinaten des Entsprechens zeigt. Für diese Tabelle werden die Koordinaten X1 und Y1 beschriftet.



- Benutzen Sie die Pfeiltasten, ◄►▲▼, um über die Tabelle zu bewegen.
- Betätigen Sie ON um zur normalen Rechneranzeige zurückzugehen.

Dieses Verfahren für das Herstellen einer Tabelle, die dieser Art Plots, kann an anderen Plotarten angewendet werden.

Plotten der Lösungen zu einfachen Differentialgleichungen

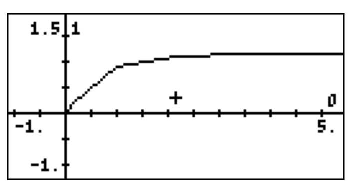
Der Plot einer einfachen Differentialgleichung kann erreicht werden, indem man `Diff Eq` auf dem `TYPE`-Gebiet des PLOT-SETUP-Klimas vorwählt, wie folgt: nehmen Sie an, daß wir x(t) von der Differentialgleichung plotten möchten dx/dt = exp(-t^2), mit Ausgangsbedingungen: x = 0 an t = 0. Der Rechner läßt das Plotten der Lösung der Differentialgleichungen von der Form Y'(T) = F(T, y) zu. Für unseren Fall ließen wir Y → x und T → t folglich F(T, y) → f(t, x) = exp(-t^2).

Bevor es die Lösung plottet, x(t), für t = 0 bis 5, löschen die Variablen EQ und PPAR.

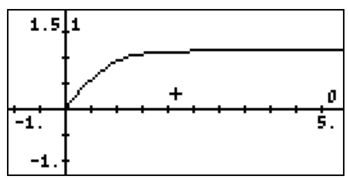
- Zunächst betätigen Sie ← 2D/3D gleichzeitig, wenn im RPN Modus, um das PLOT-SETUP-Fenster zu sehen .
- Änderne Sie `TYPE` zu `Diff Eq`.
- Betätigen Sie ▼ und tippen Sie ' ← e^x — ALPHA ← T Y^x 2 OK.
- Der Cursor ist jetzt auf dem `Indep` Gebiet. Es sollte H-Var:0 und auch V-Var:1 zeigen. Dieses ist der Code, der durch den Rechner verwendet wird, um die geplottet zu werden Variablen zu kennzeichnen. H-Var:0 bedeutet die unabhängige Variable (später vorgewählt werden) wird geplottet in der horizontalen Mittellinie. Auch V-Var:1 bedeutet die

abhängige Variable (Rückstellung Namen ' Y ') wird geplottet in der vertikalen Mittellinie.

- Betätigen Sie ▼ Der Cursor ist jetzt auf dem `Indep` Gebiet. Betätigen Sie → ' ALPHA ← T OK um das unabhängige Variable zu ändern in *t*.
- Betätigen Sie NXT OK um zur normalen Rechneranzeige zurückzugehen.
- Betätigen Sie ← WIN gleichzeitig wenn in RPN Modus, um das PLOT-Fenster zu erhalten (in diesem Fall wird er genannt PLOT WINDOW – DIFF EQ).
- Ändern Sie die H-VIEW und V-VIEW Parameter zum lesen: H-VIEW: -1 5, V-VIEW: -1 1.5
- Änderen Sie der `Init` Wert zu 0 und der abschließende Wert zu 5 durch durch das Verwenden von: 0 OK 5 OK.
- Die Werten Schritt und Tol stellen den Schritt im unabhängigen Variablen und in der Toleranz dar, damit Konvergenz durch die numerische Lösung verwendet werden kann. Lassen wir jene Werte mit ihren Standardannahmen wie sie sind (wenn die Wortrückstellung nicht im Schritt gezeigt wird: Feld, benutzen Sie NXT RESET zum Zurückstellen, der zu seinem Default-Wert bewertet. Betätigen Sie NXT um zum Hauptmenü zurückzugehen. Betätigen Sie ▼
- Der Wert Init-Soln stellt den Ausgangswert der Lösung dar, um das numerische Resultat zu beginnen. Für den anwesenden Fall haben wir für Initiale bedingen x(0) = 0, so müssen wir diesen Wert bis 0,0 ändern, indem wir 0 OK.verwenden.
- Benutzen Sie ERASE DRAW, um der Lösung zur Differentialgleichung zu plotten.
- Betätigen Sie EDIT NXT LABEL MENU um das Diagramm mit Markierung zu sehen.



- Betätigen Sie NXT um zurückzugehen im Menü. Betätigen Sie NXT PICT um das ursprüngliche graphische Menü zurückzugewinnen.
- Wenn Sie das Diagramm beobachten, das geplottet wird, sehen Sie daß das Diagramm nicht sehr glatt ist. Das ist, weil der Plotter einen Zeitschritt verwendet, der zu groß ist. Um das Diagramm zu verfeinern und es glatter zu bilden, verwenden Sie einen Schritt von 0,1. Versuchen Sie folgendes: CANCL ▼▼▼ . 1 OK ERASE DRAW Sie brauchen mehr Zeit den Plot durchzuführen, aber die Form ist sicherlich glatter als vorher.
- EDIT NXT LABEL MENU damit Sie Axtmarken und -strecke sehen. Beachten Sie, daß die Markierungen für die Äxte als 0 (Horizontal) und 1 (Vertikal).gezeigt werden. Diese sind die Definitionen für die Äxte, wie im PLOT WINDOW-Schirm (sehen Sie oben) d.h. H-VAR (t): 0, and V-VAR(x): 1.



- Betätigen Sie NXT NXT PICT um das Menü zurück zu gewinnen und zum PICT Klima zurück zu gehen.
- Betätigen Sie (X,Y) um Koordinaten irgendeines Punktes auf dem Diagramm festzustellen. Verwenden Sie ▶ und ◀ Tasten, zum verschieben des Cursors im Plotbereich. Am unteren Bildschirmrand sehen Sie die Koordinaten des Cursors wie (X,Y).). Der Rechner verwendet X und Y als die Rückstellungsnamen für beziehungsweise die horizontalen und vertikalen Äxte,.
- Betätigen Sie NXT CANCL um in dem PLOT WINDOW-Klima zurückzukehren. Betätigen Sie zunächts ON um zur normalen Rechneranzeige zurückzugehen.

Mehr Details über das Verwenden der graphischen Lösungen von Differentialgleichungen werden in Kapitel 16 dargestellt.

Wahrheit Plots

Wahrheit Plots werden benutzt um zweidimensionale Plots der Regionen zu produzieren, die einen bestimmten mathematischen Zustand erfüllen, der entweder zutreffend oder falsch sein kann `Z.B. nehmen Sie, daß Sie die Region für plotten möchten X^2/36 + Y^2/9 < 1 an` gehen Sie folgendermaßen vor:

- Zunächst betätigen Sie [←] [2D/3D] gleichzeitig, wenn im RPN Modus, um das PLOT-SETUP-Fenster zu sehen .
- Änderen Sie TYPE zu Truth.
- Betätigen Sie [▼] und tippen Sie { ' (X^2/36+Y^2/9 < 1)', '(x^2/16+y^2/9 > 1) ' } [OK] um der geplottet zu werden Bedingungen zu definieren.
- Der Cursor ist jetzt auf dem `Indep` Gebiet. Lassen Sie das als ' X ', wenn Sie bereits auf diese Variable eingestellt werden, oder ändern Sie es zu ' X ', wenn Sie benötigt werden.
- Betätigen Sie [NXT] [OK] um zur normalen Rechneranzeige zurückzugehen.
- Betätigen Sie [←] [WIN] „ò gleichzeitig wenn in RPN Modus, um das PLOT-Fenster zu erhalten (in diesem Fall wird er genannt PLOT WINDOW-TRUTH-Fenster). Lassen wir den Default-Wert für die Strecken des Fensters halten: `H-View: -6.5 6.5, V-View: -3.1 3.2` (um sie zurückzustellen brauchen Sie [NXT] [RESET] (alles auserwähltes Zurückstellen) [OK] [NXT]).

Anmerkung: Wenn die Strecken des Fensters nicht auf Default-Werte eingestellt werden, ist die schnellste Weise, sie zurückzustellen, indem Sie [NXT] [RESET] (alles auserwähltes zurückstellen) [OK] [NXT] verwenden.

- Betätigen Sie [ERASE] [DRAW] um den Wahrheit Plot zu zeichnen. Weil der Rechner das gesamte plottende Gebiet, Punkt nach Punkt probiert, dauert es einige Minuten, um einen Wahrheit Plot zu produzieren. Der anwesende Plot sollte einen schattierten Ellipse von Halbäxten 6 und 3 produzieren (beziehungsweise in x und in y,), zentriert am Ursprung.
- Betätigen Sie [EDIT] [NXT] [LABEL] [MENU] um das Diagramm mit Markierung zu sehen. Die Fensterparameter sind so, daß Sie nur Hälfte der Markierungen im X-axis sehen. Betätigen Sie [NXT] um zurückzugehen im

Menü. Betätigen Sie NXT PICT um das ursprüngliche graphische Menü zurückzugewinnen.

- Betätigen Sie (X,Y) um Koordinaten irgendeines Punktes auf dem Diagramm festzustellen. Verwenden Sie die Pfeiltasten, um den Cursor über die geplottete Region zu verschieben. Am unteren Bildschirmrand sehen Sie die Koordinaten des Cursors als (X,Y).
- Betätigen Sie NXT CANCL um in dem PLOT WINDOW-Klima zurückzukehren. Dann betätigen Sie ON oder NXT OK um zur normalen Rechneranzeige zurückzugehen.

Sie können mehr als eine Bedingung gleichzeitig plotten lassen, wenn Sie die Bedingungen multiplizieren. Z.B. um das Diagramm der Punkte zu plotten, für die $X^2/36 + Y^2/9 < 1$, und $X^2/16 + Y^2/9 > 1$, das folgende verwenden:

- Zunächst betätigen Sie ← 2D/3D gleichzeitig, wenn im RPN Modus, um das PLOT-SETUP-Fenster zu sehen .
- Betätigen Sie ▼ und tippen Sie { ' (X^2/36+Y^2/9 < 1)', '(x^2/16+y^2/9 > 1) ' } OK um der geplottet zu werden Bedingungen zu definieren.
- Betätigen Sie ERASE DRAW um den Wahrheits Plot zu zeichnen. Wieder müssen Sie geduldig sein, während der Rechner das Diagramm produziert. Wenn Sie den Plot unterbrechen möchten, betätigen Sie ON einmal. Zunächst betätigen Sie CANCL.

Histogramme, Stabplots und Streuungplots

Histogramme, Stabplots und Streuungplots werden benutzt um die getrennten Daten zu plotten, die in der reservierten Variable ΣDAT gespeichert werden. Diese Variable wird nicht nur für diese Arten der Plots, aber auch für alle Art statistische Anwendungen verwendet, wie in Kapitel 18 gezeigt wird. In Wirklichkeit wird der Gebrauch von Histogrammplots hinausgeschoben, bis wir an dieses Kapitel gelangen, denn das Plotten eines Histogramms erfordert, um eine Gruppierung von Daten und von Fourier-Analyse vor dem tatsächlichen Plot durchzuführen. In diesem Abschnitt zeigen wir, wie man Daten im variablen ΣDAT lädt und wie man Stabplots plottet und Plots zerstreut.

Wir verwenden die folgenden Daten für das Plotten der Stabplots und Streuungplots:

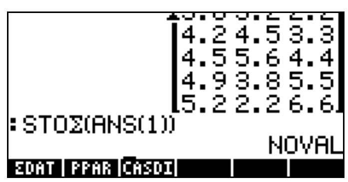
x	y	z
3.1	2.1	1.1
3.6	3.2	2.2
4.2	4.5	3.3
4.5	5.6	4.4
4.9	3.8	5.5
5.2	2.2	6.6

Stabplots

Zuerst überprüfen ob der CAS Ihres Rechners im `Exact` Modus ist. Zunächst geben Sie die Daten ein, die oben als Matrix gezeigt werden d.h.

[[3.1,2.1,1.1],[3.6,3.2,2.2],[4.2,4.5,3.3],
[4.5,5.6,4.4],[4.9,3.8,5.5],[5.2,2.2,6.6]] ENTER

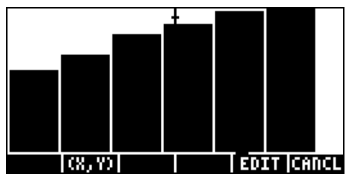
um ihn in ΣDAT zu speichern, verwenden Sie die Funktion STOΣ (vorhanden im Funktion Katalog, → CAT). Betätigen Sie VAR, um Ihr Variablen Menü zurückzugewinnen. Ein weicher MenüTaste, der als ΣDAT markiert wird, sollte im Stapel vorhanden sein. Das folgende Bild zeigt die Ablage dieser Matrix im ALG Modus:



Das Diagramm produzieren:

- Zunächst betätigen Sie ← 2D/3D gleichzeitig, wenn im RPN Modus, um das PLOT SETUP-Fenster zu sehen .
- Änderen Sie `TYPE` zu `Bar`.

- A Matrix wird am ΣDAT Feld gezeigt. Dieses ist die Matrix, die wir vorher schon in ΣDAT speicherten.
- Heben Sie der `Col` hervor: Feld Dieses Feld läßt Sie die Spalte von ΣDAT wählen, das geplottet werden soll. Der Default-Wert ist 1.
- Betätigen Sie NXT OK um zur normalen Rechneranzeige zurückzugehen.
- Zunächst betätigen Sie ← 2D/3D gleichzeitig, wenn im RPN Modus, um das PLOT WINDOW-Schirm zu sehen .
- Änderen Sie der V-VIEW um zu lesen, `V-View: 0 5`.
- Betätigen Sie ERASE DRAW um den Wahrheit Plot zu zeichnen.



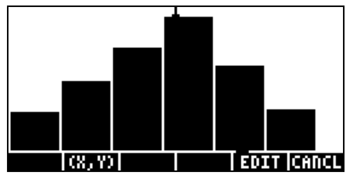
- Betätigen Sie CANCL um in dem PLOT WINDOW-Klima zurückzukehren. Dann betätigen Sie ON oder NXT OK um zur normalen Rechneranzeige zurückzugehen.

Die Zahl von den Stäben die geplottet werden stellt die Breite des Stabes fest. Das H- und V-VIEW werden durch Rückstellung eingestellt bis 10. Wir änderten die V-VIEW, um den Maximalwert in Spalte 1 von ΣDAT besser unterzubringen. Stabplots sind nützlich, wenn man kategorische (d.h., nicht numerische) Daten plotten will.

Nehmen Sie an, daß Sie die Daten in Spalte 2 der ΣDAT Matrix plotten möchten:

- Betätigen Sie ← 2D/3D gleichzeitig, wenn im RPN Modus, um das PLOT-SETUP-Fenster zu sehen .
- Betätigen Sie ▼▼ um der `Col` hervorzuheben: Feld und Art 2 OK, gefolgt von NXT OK.
- Betätigen Sie ← WIN gleichzeitig, wenn im RPN Modus, um das PLOT-SETUP-Fenster zu sehen .
- Änderen Sie der V-VIEW um `V-View` zu lesen: `0 6`

- Betätigen Sie ERASE DRAW .

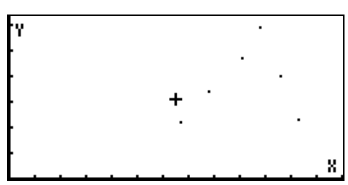


- · Betätigen Sie CANCL, um zum PLOT WINDOW-Schirm zurückzugehen, dann ON zur Rückkehr zur normalen Rechneranzeige.

Streuungplots

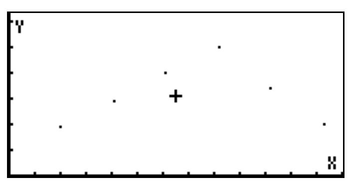
Wir benutzen die gleiche ΣDAT Matrix, um Streuungplots zu produzieren. Zuerst plotten wir die Werte von y gegen x, dann die von y gegen z, wie folgt:

- Betätigen Sie ← 2D/3D gleichzeitig, wenn im RPN Modus, um das PLOT-SETUP-Fenster zu sehen .
- Änderen Sie `TYPE` zu `Scatter`.
- Betätigen Sie ▼▼ um die `Cols` hervorzuheben: Feld. Tragen Sie 1 OK 2 OK ein, um Spalte 1 als X und Spalte 2 vorzuwählen wie Y im Y-vs.-
- Betätigen Sie NXT OK um zur normalen Rechneranzeige zurückzugehen.
- Zunächst betätigen Sie ← WIN gleichzeitig, wenn im RPN Modus, um das PLOT WINDOW-Schirm zu sehen .
- Änderen Sie das Plotfenster daß sich erstreckt, um zu lesen: H-VIEW: 0 6, V-VIEW: 0 6.
- Betätigen Sie ERASE DRAW um den Stab Plot zu zeichnen. Betätigen Sie EDIT NXT LABEL MENU, um die Plots zu sehen im Menü und mit dem Kennzeichnen der Markierungen (der Cursor jedoch ist mitten in dem Plot):



- Presse NXT NXT PICT, um der EDIT-KLIMA zu verlassen.
- Betätigen Sie CANCL um in dem PLOT WINDOW-Klima zurückzukehren. Dann betätigen Sie ON oder NXT OK um zur normalen Rechneranzeige zurückzugehen.

Um y gegen z zu plotten, verwenden Sie:

- Betätigen Sie ← 2D/3D gleichzeitig, wenn im RPN Modus, um das PLOT SETUP-Fenster zu sehen .
- Betätigen Sie ▼▼ um die `Cols` hervorzuheben: Feld. Tragen Sie 3 OK 2 OK ein, um Spalte 3 als X und Spalte 2 vorzuwählen im Y im Y-vs.-X Streuungsplot.
- Betätigen Sie NXT OK um zur normalen Rechneranzeige zurückzugehen.
- Zunächst betätigen Sie ← WIN gleichzeitig, wenn im RPN Modus, um das PLOT WINDOW-Schirm zu sehen .
- Änderen Sie das Plotfenster daß sich erstreckt, um zu lesen: H-VIEW: 0 7, V-VIEW: 0 7.
- Betätigen Sie ERASE DRAW um den Stab Plot zu zeichnen. Betätigen Sie EDIT NXT LABEL MENU, um den Plot zu sehen im Menü und mit dem Kennzeichnen der Markierungen.
- 

- Presse NXT NXT PICT, um der EDIT-KLIMA zu verlassen.
- Betätigen Sie CANCL um in dem PLOT WINDOW-Klima zurückzukehren. Dann betätigen Sie ON oder NXT OK um zur normalen Rechneranzeige zurückzugehen.

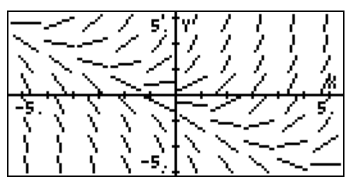
Steigungsfelder

Steigungfelder werden benutzt, um die Lösungen zu einer Differentialgleichung der Form y sichtbar zu machen ' = f(x, y). Im Allgemeinen was im Plot dargestellt wird, sind die Segmente, die zu den

Lösung Kurven, seit y ' = dy/dx tangential sind, ausgewertet an irgendeinem Punkt (x,y), darstellt die Steigung der Tangentelinie am Punkt (x,y).

Z.B. um die Lösung zur Differentialgleichung y sichtbar zu machen verwenden wir das folgende f(x, y) = x+y,:

- Betätigen Sie ↰ 2D/3D gleichzeitig, wenn im RPN Modus, um das PLOT SETUP-Fenster zu sehen .
- Änderen Sie `TYPE` zu `Slopefield`.
- Betätigen Sie ▼ und tippen Sie 'X+Y' OK.
- überprüfen, ob ' X ' als das Indep vorgewählt wird und ' Y ' als das Depnd Variablen
- Betätigen Sie NXT OK um zur normalen Rechneranzeige zurückzugehen.
- Betätigen Sie ↰ WIN gleichzeitig, wenn im RPN Modus, um das PLOTWINDOW-Schirm zu sehen .
- Änderen Sie das Plotfenster daß sich erstreckt, um zu lesen: X-Left:-5, X-Right:5, Y-Near:-5, Y-Far: 5
- Betätigen Sie ERASE DRAW um den Steigungsplot zu zeichnen. Betätigen Sie EDIT NXT LABEL MENU, um den Plot zu sehen im Menü und mit dem Kennzeichnen der Markierungen.
-

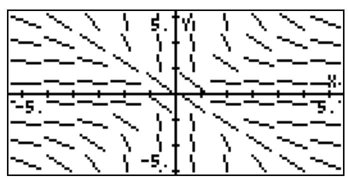


- Presse NXT NXT PICT, um der EDIT-KLIMA zu verlassen.

- Betätigen Sie CANCL um in dem PLOT WINDOW-Klima zurückzukehren. Dann betätigen Sie ON oder NXT OK um zur normalen Rechneranzeige zurückzugehen.

Wenn Sie die Steigungparzelle im Papier reproduzieren konnten, können Sie Linien eigenhändig verfolgen, die Tangente zu den Strichen sind, die im Plot gezeigt werden. Dieses zeichnet festsetzen Linien von y(x, y) = Konstante, für die Lösung von y ' = f(x, y). So sind Steigungfelder nützliche Werkzeuge für das Sichtbar machen besonders der schwierigen Gleichungen, um zu lösen.

Versuchen Sie auch mal einen Steigungsplot für Funktion y’ = f(x,y) = - $(y/x)^2$, dabei benützend:

- Betätigen Sie ← 2D/3D gleichzeitig, wenn im RPN Modus, um das PLOT-SETUP-Fenster zu sehen .
- Änderen Sie `TYPE` zu `Slopefield`.
- Betätigen Sie ▼ und tippen Sie ‘– (Y/X)^2’ OK.
- Betätigen Sie ERASE DRAW um den Steigungsplot zu zeichnen. Betätigen Sie EDIT NXT LABEL MENU, um den Plot zu sehen im Menü und mit dem Kennzeichnen der Markierungen.
- 

- Presse NXT NXT PICT, um der EDIT-KLIMA zu verlassen.
- Betätigen Sie NXT CANCL um in dem PLOT WINDOW-Schirm zurückzukehren. Dann betätigen Sie ON oder NXT OK um zur normalen Rechneranzeige zurückzugehen.

Schnelle 3D plots

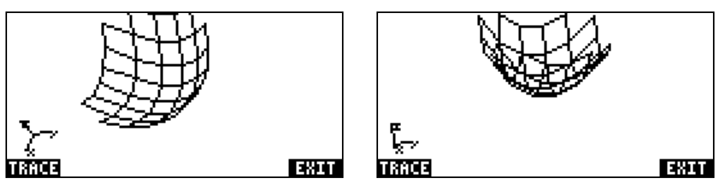
Schnelle Plots 3D werden, die dreidimensionalen Oberflächen sichtbar zu machen benutzt, die durch Gleichungen der Form z = f(x, y) dargestellt werden. Z.B. wenn Sie sichtbar machen möchten z = f(x,y) = x^2+y^2, können wir das folgende verwenden:

- Betätigen Sie ← 2D/3D gleichzeitig, wenn im RPN Modus, um das PLOT-SETUP-Fenster zu sehen .
- Änderen Sie `TYPE` zu Fast3D.
- Betätigen Sie ▼ und tippen Sie ‘X^2+Y^2’ OK.
- überprüfen, ob ' X ' als das Indep vorgewählt wird und ' Y ' als das Depnd Variablen

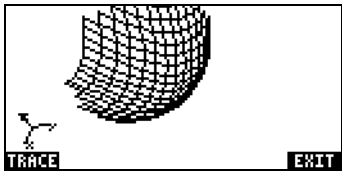
- Betätigen Sie NXT OK um zur normalen Rechneranzeige zurückzugehen.
- Zunächst betätigen Sie ↰ WIN gleichzeitig, wenn im RPN Modus, um das PLOT WINDOW-Schirm zu sehen .
- Änderen Sie das Plotfenster daß sich erstreckt, um zu lesen: X-Left:-1, X-Right:1, Y-Near:-1, Y-Far: 1, Z-Low: -1, Z-High: 1, Step Indep: 10, Depnd: 8

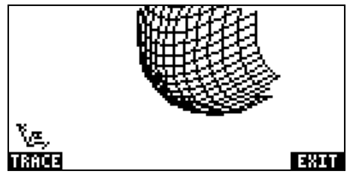
> **Anmerkung**: 1, Step Indep: und Depnd: Werte stellen Zahl dar der Rasterlinien die im Plot verwendet werden Je größere diese Nummern, um so langsamer das Diagramm zu produzieren ist, ob gleich, die Zeiten, die für graphisches Erzeugung verwendet werden, verhältnissmäßig schnell sind Vorläufig halten wir die Default-Werte von 10 und von 8 für die Schrittdaten.

- Betätigen Sie ERASE DRAW um den 3D Plot zu zeichnen. Das Resultat ist eine wireframe Abbildung der Oberfläche mit dem Bezugskoordinatesystem, das an der Ecke der untereren Linke des Schirmes gezeigt wird. Indem Sie das Pfeiltasten (◄►▲▼) verwenden, können Sie die Lagebestimmung der Oberfläche ändern. Die Lagebestimmung des Bezugskoordinatesystems ändert dementsprechend. Versuchen Sie, die Oberflächenlagebestimmung auf Ihren Selbst zu ändern. Die folgenden Abbildungen zeigen ein Paar der Ansichten des Diagramms:
-



- wenn Sie fertig sind, betätigen Sie EXIT..
- Betätigen Sie CANCL um in dem PLOT WINDOW-Klima zurückzukehren.
- Änderen Sie die Schrittdaten um zu lesen: Schritt Indep: `Step Indep: 20, Depnd: 16`
- Betätigen Sie ERASE DRAW um den Oberflächenplot zu zeichnen. Beispielansichten
-





- wenn Sie fertig sind, betätigen Sie EXIT..
- Betätigen Sie CANCL damit Sie zurückkehren zur PLOT WINDOW.
- Dann betätigen Sie ON oder NXT OK, um zur normalen Rechneranzeige zurückzugehen.

Versuch auch ein schneller Plot 3D für Oberfläche z = f(x,y) = sin (x^2+y^2)

- Betätigen Sie ← 2D/3D gleichzeitig, wenn im RPN Modus, um das PLOT-SETUP-Fenster zu sehen .
- Betätigen Sie ▼ und tippen Sie 'SIN(X^2+Y^2)' OK.
- Betätigen Sie ERASE DRAW um den Steigungsplot zu zeichnen. Betätigen Sie EDIT NXT LABEL MENU, um den Plot zu sehen im Menü und mit dem Kennzeichnen der Markierungen.
- Presse NXT NXT EXIT, um der EDIT-KLIMA zu verlassen.
- Betätigen Sie CANCL um in dem PLOT WINDOW-Klima zurückzukehren. Dann betätigen Sie ON oder NXT OK um zur normalen Rechneranzeige zurückzugehen.

Drahtframe plots

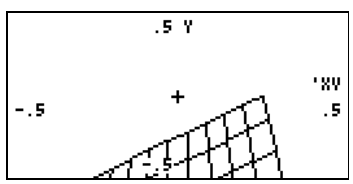
Drahtframe plots sind Plots die durch 3D Oberflächen, z = f(x, y) beschrieben werden. Anders als schnelle Plots 3D sind wireframe Plots statische Plots. Der Benutzer kann die Veranschaulichung für den Plot d.h. der Punkt wählen, von dem die Oberfläche gesehen wird. Z.B. um einen wireframe Plot für die Oberfläche zu produzieren verwenden z = x + 2y -3, das folgende:

- Betätigen Sie ← 2D/3D gleichzeitig, wenn im RPN Modus, um das PLOT-SETUP-Fenster zu sehen .
- Änderen Sie `TYPE` zu `wireframe`.
- Betätigen Sie ▼ und tippen Sie 'X+2*Y-3' OK.

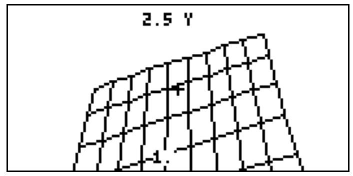
- überprüfen, ob ' X ' als das Indep vorgewählt wird und ' Y ' als das Depnd Variablen
- Betätigen Sie NXT OK um zur normalen Rechneranzeige zurückzugehen.
- Zunächst betätigen Sie ← WIN gleichzeitig, wenn im RPN Modus, um das PLOT WINDOW-Schirm zu sehen .
- Änderen Sie das Plotfenster daß sich erstreckt, um zu lesen: X-Left:-1, X-Right:1, Y-Near:-1, Y-Far: 1, Z-Low: -1, Z-Low: 1, XE:0,YE:-3, ZE:0, Step Indep: 10, Depnd: 8

Die Koordinaten XE, YE, ZE, Standplatz für "Auge Koordinaten," d.h., ., von denen die Koordinaten ein Beobachter den Plot sieht. Die gezeigten Werte sind die Default-Werte. 1, Step Indep: und Depnd: Werte stellen Zahl dar der Rasterlinien die im Plot verwendet werden Je größer der Nummer, um so langsamer das Diagramm produziert. Vorläufig halten wir die Default-Werte von 10 und von 8 für die Schrittdaten.

- Betätigen Sie ERASE DRAW um den 3D Plot zu zeichnen. Das Resultat ist eine Drahtframe Abbildung der Oberfläche.
- Betätigen Sie EDIT NXT LABEL MENU um das Diagramm mit Markierung und Strecke zu sehen. Diese bestimmte Version des Diagramms wird auf das untere Teil der Anzeige begrenzt. Wir können die Veranschaulichung ändern, um eine andere Version des Diagramms zu sehen.

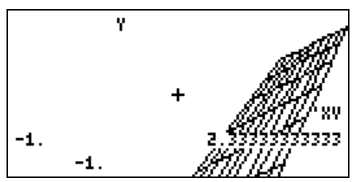


- Betätigen Sie NXT NXT PICT CANCL um in dem PLOT WINDOW-Klima zurückzukehren.
- ·Änderen Sie die Augenkoordinatdaten um zu lesen: `XE:0 YE:-3 ZE:3`
- Betätigen Sie ERASE DRAW um den Oberflächenplot zu zeichnen.
- Betätigen Sie EDIT NXT LABEL MENU um das Diagramm mit Markierung und Strecke zu sehen.



Diese Version des Diagramms besetzt mehr Bereich in der Anzeige als die vorhergehende. Wir können die Veranschaulichung ändern, um eine andere Version des Diagramms zu sehen.

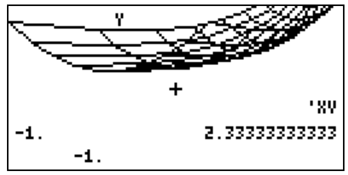
- Betätigen Sie NXT NXT PICT CANCL um in dem PLOT WINDOW-Klima zurückzukehren.
- Änderen Sie die Augenkoordinatdaten um zu lesen: XE:3 YE:3 ZE:3
- Betätigen Sie ERASE DRAW um den Oberflächenplot zu zeichnen. Dieses Mal sitzt der Hauptteil des Plots nach rechts in der Anzeige.



- Betätigen Sie CANCL um in dem PLOT WINDOW-Klima zurückzukehren.
- Dann betätigen Sie ON oder NXT OK, um zur normalen Rechneranzeige zurückzugehen.

Versuch auch ein Wireframe Plot für Oberfläche z = f(x,y) = sin (x^2+y^2)

- Betätigen Sie ← 2D/3D gleichzeitig, wenn im RPN Modus, um das PLOT-SETUP-Fenster zu sehen .
- Betätigen Sie ▼ und tippen Sie ‘X^2+Y^2’ OK.
- Betätigen Sie ERASE DRAW um den Steigungsplot zu zeichnen. Betätigen Sie EDIT NXT MENU LABEL um den Plot zu sehen im Menü und mit dem Kennzeichnen der Markierungen.

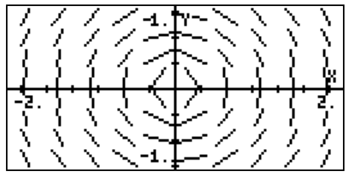


- Presse NXT NXT PICT, um der EDIT-KLIMA zu verlassen.
- Betätigen Sie CANCL um in dem PLOT WINDOW-Klima zurückzukehren. Dann betätigen Sie ON oder NXT OK um zur normalen Rechneranzeige zurückzugehen.

Ps-Kontour plots:

Ps-Kontour plots sind Plots die durch 3D Oberflächen, z = f(x, y) beschrieben werden. Die produzierten Formen sind Projektionen der waagerecht ausgerichteten Oberflächen z = Konstante auf der x-y Fläche. Z.B. um einen Ps-Kontour Plot für die Oberfläche zu produzieren verwenden $z = x^2+y^2$, das folgende:

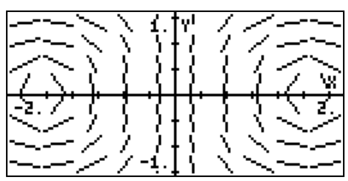
- Betätigen Sie ← 2D/3D gleichzeitig, wenn im RPN Modus, um das PLOT-SETUP-Fenster zu sehen .
- · Ändern Sie TYPE zur Ps-Kontour.
- Betätigen Sie ▽ und tippen Sie 'X^2+Y^2' OK.
- überprüfen, ob ' X ' als das Indep vorgewählt wird und ' Y ' als das Depnd Variablen
- Betätigen Sie NXT OK um zur normalen Rechneranzeige zurückzugehen.
- Zunächst betätigen Sie ← WIN gleichzeitig, wenn im RPN Modus, um das PLOT WINDOW-Schirm zu sehen .
- Änderen Sie das Plotfenster daß sich erstreckt, um zu lesen: X-Left:-2, X-Right:2, Y-Near:-1, Y-Far: 1, Step Indep: 10, Depnd: 8
- Betätigen Sie ERASE DRAW um den Kontourplot zu zeichnen. Dieser Betrieb dauert einige Zeit also Geduld. Das Resultat ist eine Konoutplor der Oberfläche. Beachten Sie, daß die Form nicht notwendigerweise ununterbrochen sind, jedoch sie eine gute Abbildung der waagerecht ausgerichteten Oberflächen der Funktion liefern.
- Betätigen Sie EDIT NXT LABEL MENU um das Diagramm mit Markierung und Strecke zu sehen.



- Betätigen Sie NXT NXT PICT CANCL um in dem PLOT WINDOW-Klima zurückzukehren.
- Dann betätigen Sie ON oder NXT OK, um zur normalen Rechneranzeige zurückzugehen.

Versuch auch ein Ps-Kontour Plot für Oberfläche z = f(x,y) = sin x cos y.

- Betätigen Sie ← 2D/3D gleichzeitig, wenn im RPN Modus, um das PLOT-SETUP-Fenster zu sehen .
- Betätigen Sie ▼ und tippen Sie ‘SIN(X)*COS(Y)’ OK.
- Betätigen Sie ERASE DRAW um den Steigungsplot zu zeichnen. Betätigen Sie EDIT NXT LABEL MENU um den Plot zu sehen im Menü und mit dem Kennzeichnen der Markierungen.
-

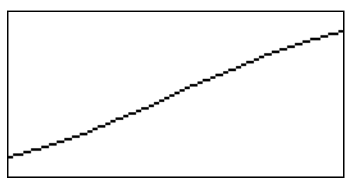


- Presse NXT NXT PICT, um der EDIT-KLIMA zu verlassen.
- Betätigen Sie CANCL um in dem PLOT WINDOW-Klima zurückzukehren. Dann betätigen Sie ON oder NXT OK um zur normalen Rechneranzeige zurückzugehen.

Y-Scheibe plots

Y-Scheibe Plots sind lebhafte Plots von z-vs.-y für unterschiedliche Werte von x von der Funktion z = f(x, y). Z.B. um einen Y-Scheibe Plot für die Oberfläche zu produzieren verwenden $z = x^3+y^3$, das folgende:

- Betätigen Sie ⬅ 2D/3D gleichzeitig, wenn im RPN Modus, um das PLOT-SETUP-Fenster zu sehen .
- · Ändern Sie TYPE zu Y-Slice.
- Betätigen Sie ▼ und tippen Sie 'X^3+X*Y^3' OK.
- überprüfen, ob ' X ' als das Indep vorgewählt wird und ' Y ' als das Depnd Variablen
- Betätigen Sie NXT OK um zur normalen Rechneranzeige zurückzugehen.
- Zunächst betätigen Sie ⬅ WIN gleichzeitig, wenn im RPN Modus, um das PLOT WINDOW-Schirm zu sehen .
- Änderen Sie das Plotfenster daß sich erstreckt, um zu lesen: X-Left:-1, X-Right:1, Y-Near:-1, Y-Far: X-Left:-1, X-Right:1, Y-Near:-1, Y-Far: 1, Z-Low:-1, Z-High:1, Schritt Indep: 10, Depnd: 8
- Betätigen Sie ERASE DRAW um den 3D Plot zu zeichnen. Sie sehen den Rechner, eine Reihe Kurven auf dem Schirm, der zu produzieren verschwinden sofort. Sie sehen den Rechner, eine Reihe Kurven auf dem Schirm, der zu produzieren verschwinden sofort. Eine der Kurven wird unten gezeigt.



- Betätigen Sie ON um die Animation zu annullieren. Betätigen Sie CANCL um in dem PLOT WINDOW-Klima zurückzukehren.
- Dann betätigen Sie ON oder NXT OK, um zur normalen Rechneranzeige zurückzugehen.

Versuch auch ein Ps-Kontour Plot für Oberfläche z = f(x,y) = (x+y) sin y.

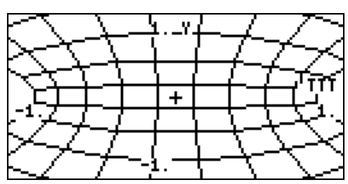
- Betätigen Sie ⬅ 2D/3D gleichzeitig, wenn im RPN Modus, um das PLOT SETUP-Fenster zu sehen .
- Betätigen Sie ▼ und tippen Sie '(X+Y)*SIN(Y)' OK.
- Presse ERASE DRAW, um die Y-Scheibe Animation zu produzieren
- Betätigen Sie ON um die Animation zu annullieren.

- Betätigen Sie CANCL um in dem PLOT WINDOW-Klima zurückzukehren. Dann betätigen Sie ON oder NXT OK um zur normalen Rechneranzeige zurückzugehen.

Gridmap plots

Gridmap Plots produzieren ein Rasterfeld der orthogonalen Kurven, die eine Funktion einer komplexen Größe der Form W = f(z) = f(x+iy) beschreiben, wo z = x+iy eine komplexe Größe ist. Die geplotteten Funktionen entsprechen dem realen und Imaginärteil von W = F(x, y) + iY(x, y), d.h. stellen sie Kurven F(x, y) = Konstante und Y(x, y) = Konstante dar.Z.B. um einen Gridmap Plot für die Funktion zu produzieren verwenden W = sin(z), das folgende:

- Betätigen Sie ← 2D/3D gleichzeitig, wenn im RPN Modus, um das PLOT SETUP-Fenster zu sehen .
- Änderen Sie TYPE zu Gridmap.
- Betätigen Sie ▼ und tippen Sie 'SIN(X+i*Y)' OK.
- überprüfen, ob ' X ' als das Indep vorgewählt wird und ' Y ' als das Depnd Variablen
- Betätigen Sie NXT OK um zur normalen Rechneranzeige zurückzugehen.
- Zunächst betätigen Sie ← WIN gleichzeitig, wenn im RPN Modus, um das PLOT WINDOW-Schirm zu sehen .
- Änderen Sie das Plotfenster daß sich erstreckt, um zu lesen: X-Left:-1, X-Right:1, Y-Near:-1 Y-Far: 1, XXLeft:-1 XXRight:1, YYNear:-1, yyFar: 1, Step Indep: 10 Depnd: 8
- Betätigen Sie ERASE DRAW um den Gridmap Plot zu zeichnen. Das Resultat ist ein Rasterfeld der Funktionen, die den realen und Imaginärteilen der komplizierten Funktion entsprechen.
- Betätigen Sie EDIT NXT LABEL MENU um das Diagramm mit Markierung und Strecke zu sehen.



- Betätigen Sie NXT NXT PICT CANCL um in dem PLOT WINDOW-Klima zurückzukehren.

- Dann betätigen Sie ON oder NXT OK, um zur normalen Rechneranzeige zurückzugehen.

Andere Funktionen von solchen kompliezierten Variablen sind:

(1) SIN((X,Y))	d.h., F(z) = sin(z)	(2)(X,Y)^2	d.h., F(z) = z^2
(3) EXP((X,Y))	d.h., F(z) = e^z	(4) SINH((X,Y))	d.h., F(z) = sinh(z)
(5) TAN((X,Y))	d.h., F(z) = tan(z)	(6) ATAN((X,Y))	d.h., F(z) = $\tan^{-1}(z)$
(7) (X,Y)^3	d.h., F(z) = z^3	(8) 1/(X,Y)	d.h., F(z) = 1/z
(9) √ (X,Y)	d.h., F(z) = $z^{1/2}$		

Parametrische-Oberfläche plots:

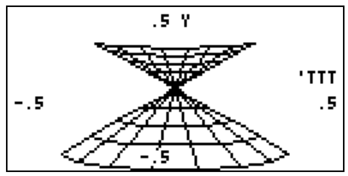
Parametrische Oberfläche Plots werden, eine dreidimensionale Oberfläche zu plotten benutzt deren Koordinaten (x, y, z) werden durch x = x(X, Y), y = y(X, Y), z=z(X, Y) beschrieben, wo X und Y unabhängige Parameter sind.

Anmerkung: Die Gleichungen x = x(X,Y), y = y(X,Y), z=z(X,Y) stellen eine parametrische Beschreibung einer Oberfläche dar. X and Y sind die unabhängigen Parameter. Die meisten Lehrbücher verwenden (u,v) als die Parameter, anstatt (X,Y). So wird die parametrische Beschreibung einer Oberfläche als x = x(u,v), y = y(u,v), z=z(u,v) gegeben.

Z.B. um einen Fotorezeptor-Oberfläche Plot für die Oberfläche x zu produzieren = verwenden x(X, Y)- = X Sünde Y, y = y(X, Y) = x Lattich Y, z=z(X, Y)=X, das folgende:

- Betätigen Sie ← 2D/3D gleichzeitig, wenn im RPN Modus, um das PLOT SETUP-Fenster zu sehen .
- · Ändern Sie TYPE zur Pr-Surface.
- Betätigen Sie ▼ und tippen Sie '{X*SIN(Y), X*COS(Y), X}' OK.
- überprüfen, ob ' X ' als das Indep vorgewählt wird und ' Y ' als das Depnd Variablen
- Betätigen Sie NXT OK um zur normalen Rechneranzeige zurückzugehen.
- Zunächst betätigen Sie ← WIN gleichzeitig, wenn im RPN Modus, um das PLOT WINDOW-Schirm zu sehen .

- Änderen Sie das Plotfenster daß sich erstreckt, um zu lesen: X-Left:-1, X-Right:1, Y-Near:-1, Y-Far: 1, Z-Low: -1, Z-High:1, XE: 0, YE:-3, zE:0, Step Indep: 10, Depnd: 8
- Betätigen Sie ERASE DRAW um den 3D Plot zu zeichnen.
- Betätigen Sie EDIT NXT LABEL MENU um das Diagramm mit Markierung und Strecke zu sehen.



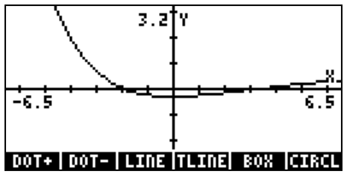
- Betätigen Sie NXT NXT PICT CANCL um in dem PLOT WINDOW-Klima zurückzukehren.
- Dann betätigen Sie ON oder NXT OK, um zur normalen Rechneranzeige zurückzugehen.

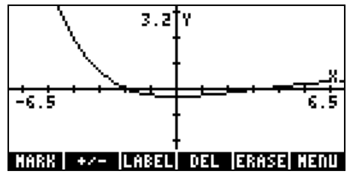
Das VPAR Variable

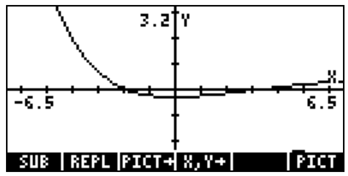
Die VPAR (Volumen-Parameter) Variable enthält die Informationen betreffend sind das "Volumen", das benutzt wird, um ein dreidimensionales Diagramm zu produzieren. Folglich sehen Sie, daß sie produzierte, wann immer Sie einen dreidimensionalen Plot wie Fast3D, Wireframe oder Fotorezeptor-Oberfläche verursachen.

Interaktieve Zeichnung

Wann immer wir ein zweidimensionales Diagramm produzieren, finden wir in den Graphiken aussortieren einen weichen MenüTaste, der EDIT beschriftet wird. Das Betätigen von EDIT produziert ein Menü, die die folgenden Wahlen einschließen (die Presse NXT, zum der zusätzlichen Funktionen zu sehen):







Durch die Beispiele oben, haben Sie die Gelegenheit, Funktionen LABEL, MENU, PICTà und REPL auszuprobieren. Viele der restlichen Funktionen, wie DOT+, DOT-, können LINE, BOX, CIRCL, MARK, DEL, etc., verwendet werden, um Punkte, Linien, Kreise, etc. auf dem Graphikschirm zu zeichnen, wie unten beschrieben. Um zu sehen wie man diese Funktionen wir versucht die folgende Übung verwendet:

Zuerst erhalten wir den Graphikschirm, der den folgenden Anweisungen entspricht:

- Betätigen Sie [←] [2D/3D] gleichzeitig, wenn im RPN Modus, um das PLOT SETUP-Fenster zu sehen .
- ·Änderung `TYPE` zur `Function`, wenn es benötigt ist.
- ·Änderung EQ an ' X '
- · überprüfen, ob Indep: wird auf ' X ' auch eingestellt
- Betätigen Sie [NXT] [OK] um zur normalen Rechneranzeige zurückzugehen.
- Betätigen Sie [←] [WIN] gleichzeitig wenn in RPN Modus, um das PLOT-Fenster zu erhalten (in diesem Fall wird er genannt PLOT-POLAR Fenster).
- · Ändern Sie die H-VIEW Strecke bis -10 bis 10, indem Sie [1] [0] [+/-] [OK] [1] [0] [OK], und die V-VIEW verwenden, Strecke bis -5 bis 5, indem [5] [+/-] [OK] [5] [OK].verwenden.
- Betätigen Sie [ERASE] [DRAW] um die Funktion zu plotten.

- Betätigen Sie EDIT NXT LABEL um das zu Diagramm markieren. Betätigen Sie NXT NXT (or ← PREV) um das ursprüngliche EDIT Menü zurückzugewinnen.

Zunächst veranschaulichen wir den Gebrauch von den unterschiedlichen zeichnenden Funktionen auf dem resultierenden Graphikschirm. Sie erfordern Gebrauch von dem Cursor und den Pfeiltasten (◀ ▶ ▲ ▼) den Cursor über den Graphikschirm zu verschieben.

DOT + und DOT-

Wenn DOT+ vorgewählt wird, werden Pixel aktiviert, wohin der Cursor das Verlassen hinter einer Spur der Cursor-Position verschiebt. Wenn DOT- vorgewählt wird, tritt der gegenüberliegende Effekt auf, werden d.h. . da Sie den Cursor verschieben, Pixel gelöscht.

Z.B. verwenden Sie ▶ ▲ die Taste, zum des Cursors mitten in dem ersten Quadranten der x-y Fläche irgendwo zu verschieben, betätigen dann DOT+ Die Markierung wird vorgewählt (DOT+■). Betätigen Sie und halten Sie den ▶ Taste, um eine horizontale Linie zu sehen, die verfolgt wird. Jetzt Presse DOT-, diese Wahl DOT-■ vorwählen. Betätigen Sie und halten Sie den ◀ Taste, um die Linie zu sehen, die Sie gerade gelöscht werden verfolgten. Presse DOT-, wenn Sie fertig sind, um diese Wahl abzuwählen.

MARK

Dieser Befehl erlaubt dem Benutzer, einen Markierung Punkt, der für eine Anzahl von Zwecken benutzt werden kann, wie einzustellen:

- Anfang der Linie mit dem LINE oder TLINE Befehl
- Ecke für einen BOX-Befehl
- Mitte für einen CIRCL-Befehl

Das Verwenden selbst des MARK Befehls läßt einfach ein x in der Position der Markierung (die Presse NXT MARK, zum sie in der Tätigkeit zu sehen).

LINE

Dieser Befehl wird, eine Linie zwischen zwei Punkte im Diagramm zu zeichnen verwendet. Um ihn in der Tätigkeit zu sehen, bringen Sie den Cursor irgendwo in den ersten Quadranten, in Position und betätigen Sie ← PREV LINE. Eine MARK wird über den Cursor gesetzt, der den Ursprung von anzeigt Verwenden Sie den ▶ Taste, um den Cursor auf der rechten Seite der gegenwärtigen Position zu verschieben, sagen Sie ungefähr 1 Zentimeter rechts, und betätigen Sie LINE.. Eine Linie ist abgehobener Betrag zwischen dem ersten und den Letztpunkten.

Beachten Sie, daß der Cursor am Ende dieser Linie noch aktiv ist, anzeigend, daß der Rechner bereit ist, eine Linie zu plotten, die an diesem Punkt beginnt. Betätigen Sie ▼ um den Cursor abwärts zu verschieben, sagen Sie über einen anderen Zentimeter, und betätigen Sie LINE wieder. Jetzt sollten Sie einen geraden Winkel durch horizontale und einer Vertikale Segmente verfolgen lassen. Der Cursor ist noch aktiv. Ihn, ohne ihn an allen zu verschieben zu entaktivieren, betätigen LINE.. Der Cursor geht zu seiner normalen Form (ein Kreuz) zurück und die LINIE Funktion ist nicht mehr aktiv.

TLINE

(Toggle-LINIE) verschieben Sie den Cursor auf den zweiten Quadranten, um diese Funktion in der Tätigkeit zu sehen. Betätigen Sie TLINE. Eine MARK wird am Anfang der Toggle-Linie gesetzt. Verschieben Sie den Cursor mit den Pfeiltasten weg von diesem Punkt, und betätigen Sie TLINE..Eine Linie wird von der gegenwärtigen Cursor-Position zum Bezugspunkt gezeichnet, der früh vorgewählt wird. Pixel, die eingeschaltet im Linie Weg sind, werden weg und umgekehrt gedreht. Die neueste Linie zu entfernen verfolgte, betätigt TLINE wieder. Um TLINE zu entaktivieren, verschieben Sie den Cursor auf den ursprünglichen Punkt in dem TLINE aktiviert wurde, und Presse LINE LINE.

BOX

Dieser Befehl wird, einen Kasten im Diagramm zu zeichnen verwendet. Verschieben Sie den Cursor auf einen freien Bereich des Diagramms, und betätigen Sie BOX. Dieses hebt den Cursor hervor. Verschieben Sie den Cursor mit den Pfeiltasten auf einen Punkt weg und in eine diagonale Richtung, von der gegenwärtigen Cursor-Position. Betätigen Sie BOX wieder. Ein Viereck wird gezeichnet dessen Diagonale die Initiale und Ende-Cursor-Positionen verbindet. Die Ausgangsposition des Kastens wird noch mit und x gekennzeichnet. Das Verschieben des Cursors auf eine andere Position und das Betätigen von von BOX erzeugen einen neuen Kasten, der den Ausgangspunkt enthält. Um BOX abzuwählen, verschieben Sie den Cursor auf den ursprünglichen Punkt in dem BOX aktiviert wurde, LINE LINE dann zu betätigen.

CIRCL

Dieser Befehl produziert einen Kreis. Kennzeichnen Sie die Mitte des Kreises mit einem MARK Befehl, dann verschieben Sie den Cursor auf einen Punkt, der ein Teil der Peripherie des Kreises ist, und betätigen Sie CIRCL. Um CIRCL zu entaktivieren, bringen Sie den Cursor zur MARK Position zurück und betätigen Sie LINE..

Versuchen Sie diesen Befehl, indem Sie den Cursor auf ein freies Teil des Diagramms verschieben, betätigen Sie MARK.. Verschieben Sie den Cursor auf einen anderen Punkt, dann betätigen Sie CIRCL.. Ein Kreis, der an der MARK und am Überschreiten durch den letzten Punkt zentriert wird, wird gezeichnet.

LABEL

Das Betätigen von LABEL legt die Label in das x und in die y-axes des gegenwärtigen Plots. Diese Eigenschaft ist weitgehend durch dieses Kapitel benutzt worden.

DEL

Dieser Befehl wird, Teile des Diagramms zwischen zwei MARK Positionen zu entfernen verwendet. Verschieben Sie den Cursor auf einen Punkt im Diagramm, und betätigen Sie MARK Verschieben Sie den Cursor auf einen anderen Punkt, betätigen Sie MARK wieder. Dann Presse DEL. Der Abschnitt des Diagramms, das zwischen den zwei Markierungen geschachtelt wird, wird gelöscht.

ERASE
Die Funktion ERASE freie Räume das gesamte Graphikfenster. Dieser Befehl ist im PLOT-Menü, sowie in den plottenden Fenstern vorhanden, die durch die weichen MenüTaste zugänglich sind.

MENU:
Das Betätigen von von MENU entfernt die frei belegbare Funktionstaste Menüaufkleber, um die Graphik zu zeigen unencumbered durch jene Aufkleber. Um die Aufkleber zurückzugewinnen, betätigen Sie NXT.

SUB
Der extrahierte Gegenstand wird automatisch in den Stapel gelegt. Wählen Sie die Teilmenge, die Sie extrahieren möchten, indem Sie eine MARK an einem Punkt in das Diagramm legen vor und den Cursor auf die diagonale Ecke des Viereckes verschieben, welches die Graphikteilmenge umgibt, und betätigen Sie SUB. Diese Eigenschaft kann benutzt werden, um Teile von zu verschieben Graphiken einwenden um das Diagramm.

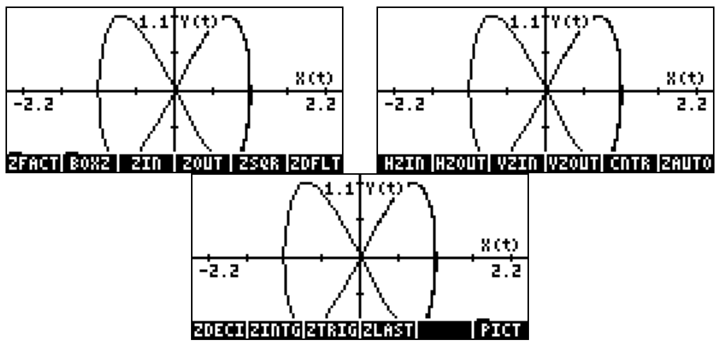
REPL
Dieser Befehl legt den Inhalt eines graphischen Gegenstandes z.Z. in Stapelniveau 1 am Ort des Positionsanzeigers im Graphikfenster. Die obere linke Ecke des graphischen Gegenstandes, der im Diagramm eingesetzt wird, wird in der Cursor-Position gesetzt.

PICT→
Dieser Befehl legt eine Kopie des Diagramms z.Z. in das Graphikfenster an zum Stapel als graphischer Gegenstand Der graphische Gegenstand, der in den Stapel gelegt wird, kann in einen variablen Namen für Ablage oder andere Art Handhabung gespeichert werden.

X,Y→
Dieser Befehl kopiert die Koordinaten der gegenwärtigen Cursor-Position, in den Benutzerkoordinaten, im Stapel.

ZOOMING IN UND AUS

Wann immer Sie eine zweidimensionale FUNCTION Graphik wechselwirkend produzieren, läßt der erste Weichmenü Taste, beschriftet ZOOM, Sie Funktionen zugänglich machen, die zoom verwendet werden können innen und heraus in die gegenwärtige Graphikanzeige. Das ZOOM-Menü umfaßt die folgenden Funktionen (die Presse, zum auf das folgende Menü zu bewegen):



Wir stellen jedes diese Funktionen des Folgens dar. Sie gerade Notwendigkeit, ein Diagramm zu produzieren, wie in Kapitel 12 oder mit einem der Programme angezeigt verzeichneten früh in diesem Kapitel.

ZFACT, ZIN , ZOUT , ZLAST

Das Betätigen von ZFACT produziert einen Eingang Schirm, der Ihnen erlaubt, das gegenwärtige x und die Y-Faktoren zu ändern. Das x und die Y-Faktoren beziehen die horizontalen und vertikalen verbraucherbestimmten Maßeinheit Strecken auf ihren entsprechenden Pixelstrecken. Ändern Sie den H-Faktor, um 8. zu lesen, und betätigen Sie OK, dann ändern Sie den V-Faktor, um 2. zu lesen, und betätigen Sie OK, Überprüfen Sie weg die Wahl ✓`Recenter on cursor` und Presse OK.

Ziehen Sie in der Graphikanzeige zurück, betätigen Sie ZIN. Die Graphik wird mit den neuen vertikalen und horizontalen Normierungsfaktoren Re-gezeichnet, zentriert in der Position, in der der Cursor, beim Beibehalten der ursprünglichen PICT Größe lokalisiert wurde (d.h., die ursprüngliche Zahl

Pixeln in beiden Richtungen). Mit den Pfeiltasten scroll horizontal oder vertikal, insoweit Sie von zoomed- im Diagramm können.

Zoom heraus, unterworfen dem h und den V-Faktoren stellte mit ZFACT, betätigen ZOOM ZOUT ein. Das resultierende Diagramm liefert mehr Detail als zoomed- im Diagramm. Sie können zum allerletzten Zoomfenster immer zurückkommen, indem Sie ZLAST verwenden.

BOXZ

Zooming in und aus einem gegebenen Diagramm kann durchgeführt werden, indem man den Weichmenü Taste BOXZ verwendet. Mit BOXZ, das Sie den rechteckigen Sektor (den "Kasten") diesen Sie vorwählen, möchten zoom innen in. Verschieben Sie den Cursor bis eine der Ecken des Kastens (die Pfeiltasten verwendend), und betätigen Sie ZOOM BOXZ. Mit den Pfeiltasten noch einmal, verschieben Sie den Cursor auf die gegenüberliegende Ecke des gewünschten Zoomkastens. Der Cursor verfolgt den Zoomkasten auf dem Schirm. Wenn gewünschter Zoomkasten vorgewählt wird, betätigen Sie ZOOM. Der Rechner zoom in den Inhalt des Zoomkastens, den Sie vorwählten, um den gesamten Schirm zu füllen.

Wenn Sie jetzt ZOUT betätigen, zoom der Rechner aus dem gegenwärtigen Kasten heraus mit dem H- und den V-Faktoren, die möglicherweise nicht die Diagrammansicht zurückgewinnen können, von der Sie den Zoomkastenbetrieb begannen.

ZDFLT, ZAUTO

Das Betätigen von von ZDFLT re-draws den gegenwärtigen Plot mit der Rückstellung x und Ystrecken d.h. -6.5 bis 6.5 in x und -3.1 bis 3.1 in y. Der Befehl ZAUTO stellt andererseits ein Zoomfenster mit der gegenwärtigen unabhängigen Variable (X) Strecke her, aber, die abhängige Variable (Y) Strecke justierend, um die Kurve zu passen (als, wenn Sie verwenden, die Funktion geben AUTO im PLOT WINDOW Form ein (← WIN "ò,gleichzeitig im RPN Modus).

HZIN, HZOUT, VZIN, VZOUT

Diese Funktionen zoom auf und aus dem Graphikschirm in der horizontalen oder vertikalen Richtung entsprechend dem gegenwärtigen h und den V-Faktoren.

CNTR

Zooms innen mit der Mitte des Zoomfensters im gegenwärtigen Ort des Positionsanzeigers. Die zooming verwendeten Faktoren sind das gegenwärtige H- und die V-Faktoren.

ZDECI

Zooms das Diagramm, damit weg von den Begrenzungen auf den Xabstand zu einem dezimalen Wert zu runden.

ZINTG

Zooms das Diagramm, damit die Pixelmaßeinheiten Benutzer-definieren Maßeinheiten werden. Z.B. hat das Fenster des Minimums PICT 131 Pixel. Wenn Sie ZINTG verwenden, mit dem Cursor in der Mitte des Schirmes, erhält das Fenster zoomed, damit der X-axis von -64.5 bis 65.5 verlängert.

ZSQR

Zooms das Diagramm, damit die plottende Skala bei 1:1 beibehalten wird, indem man die x Skala justiert und hält die y Skala repariert, wenn das Fenster breiter als höher ist. Dieses zwingt proportionales zooming.

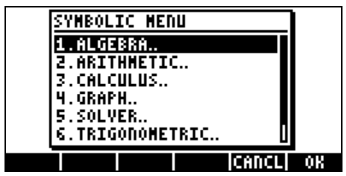
ZTRIG

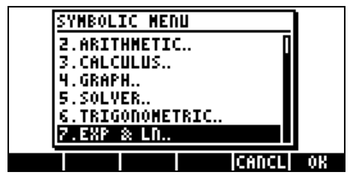
Zooms das Diagramm, damit die x Skala eine Strecke von ungefähr -3π zu $+3\pi$ enthält, die bevorzugte Strecke für trigonometrische Funktionen.

Anmerkung: Keine dieser Funktionen sind programmierbar. Sie sind in einer wechselwirkenden Weise nur nützlich. Verwirren Sie den Befehl ZFACT im ZOOM-Menü nicht mit der Funktion ZFACTOR, die für das dynamische Gas und die Chemieanwendungen verwendet wird (sehen Sie Kapitel 3).

Das SYMBOLIC Menü und Diagramme

Das SYMBOLIC Menü wird durch das Betätigen des SYMB Tastes aktiviert (vierter Taste vom links in der vierten Reihe von der Oberseite der Tastatur). Dieses Menü liefert eine Liste der Menüs, die auf dem Computer-algebraischen System bezogen werden, oder CAS, diese sind:





Alle als eins dieser Menüs sind direkt in der Tastatur vorhanden, indem sie wie folgt die passende Tastenanschlagkombination betätigen. Das Kapitel des Benutzerhandbuches, in dem die Menüs beschrieben werden, wird auch verzeichnet:

ALGEBRA	↱ ALG (the 4 Taste)	Ch. 5
ARITHMETIC..	↰ ARITH (the 1 Taste)	Ch. 5
CALCULUS..	↰ CALC (the 4 Taste)	Ch. 13
SOLVER..	↰ S.SLV (the 7 Taste)	Ch. 6
TRIGONOMETRIC..	↱ TRIG (the 8 Taste)	Ch. 5
EXP&LN..	↰ EXP&LN (the 8 Taste)	Ch. 5

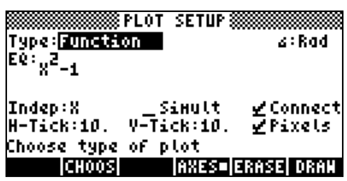
SYMB/GRAPH Menü

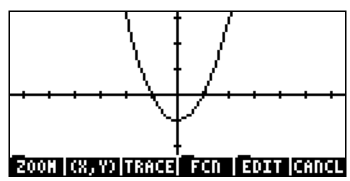
Das DIAGRAMM-Unterprogramm innerhalb des SYMB Menüs schließt die folgenden Funktionen ein:

DEFINE: selben wie die Tastenanschlagreihenfolge DEF (the 2 Taste)
GROBADD: Pasten zwei GROBs zuerst Überschuß die Sekunde (sehen Sie Kapitel 22)
PLOT (Funktion): plottet eine Funktion, ähnliches 2D/3D
PLOT (Funktion): fügt diese Funktion der Liste von Funktionen Plot hinzu, ähnlich 2D/3D
Plot setup: selben wie 2D/3D
SIGNTAB(function): SIGNTAB(function): unterzeichnen Sie Tabelle der gegebenen Funktion, die Abstände der positiven und negativen Veränderung, der nullpunkte und der endlosen Asymptotes zeigt
TABVAL: Tabelle der Werte für eine Funktion
TABVAR: Veränderung Tabelle einer Funktion

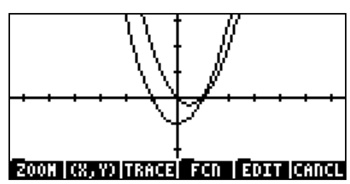
Beispiele von einigen dieser Funktionen werden zunächst zur Verfügung gestellt.

PLOT(X^2-1) ist ähnliches 2D/3D mit EQ: . X^2 -1. Das Verwenden von von ERASE DRAW produziert den Plot:



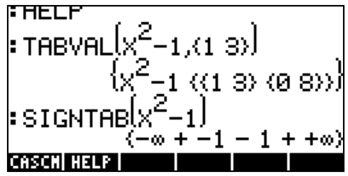


PLOTADD(X^2-X) ist ähnliches 2D/3D aber Hinzufügen dieser Funktion EQ: X^2 -1. Das Verwenden von ERASE DRAW produziert den Plot:



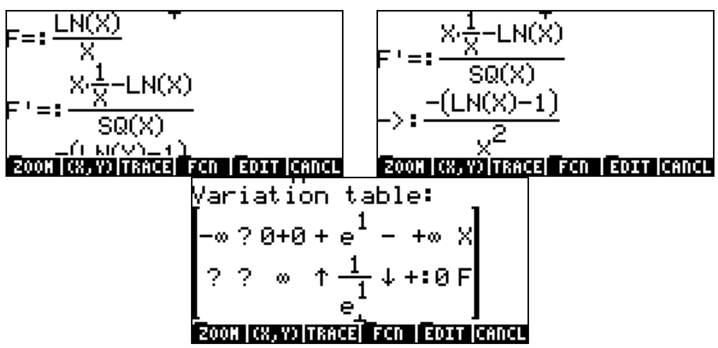
TABVAL(X^2-1,{1, 3}) produziert eine Liste { Min - Max } der Werte Maximum der Funktion im Abstand { 1,3}, während SIGNTAB(X^2-1) das Zeichen der

Funktion im Abstand (-∞,+), mit f(x) 0 > innen (-∞,-1) zeigt, f(x) < 0, innen (-1.1) und f(x) > 0 innen (1,+ ∞).



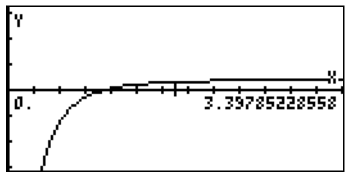
TABVAR(LN(X)/X) produziert die folgende Tabelle der Veränderung:

Eine ausführliche Deutung der Tabelle der Veränderung ist einfacher, in RPN Modus zu folgen:



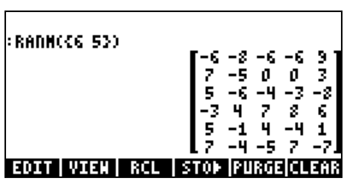
Der Ausgang ist in einem graphischen Format und zeigt die ursprüngliche Funktion, F(X), die Ableitung F'(X) nach rechts nach Ableitung und nach Vereinfachung und schließlich eine Tabelle der Veränderung. Die Tabelle besteht aus zwei Reihen, beschriftet in der rechten Seite. So stellt die obere Reihe Werte von X dar und die zweite Reihe stellt Werte von F dar. Die Fragezeichen zeigt Ungewißheit oder Nichtdefinition an. Z.B. für X<0, wird

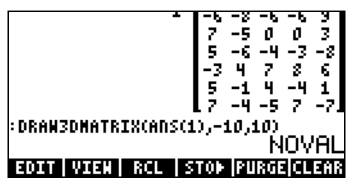
LN(X) nicht definiert, so zeichnet das X Erscheinen ein Fragezeichen in diesem Abstand. . Recht an null (0+0) F, für ist X = e, F = 1/e endlos. F erhöht sich vor dem Erreichen dieses Wertes, wie angezeigt durch den aufwärts Pfeil, und den Abnahmen nach diesem Wert (X=e) werden etwas größer als null (+:0) wie X zur Unbegrenztheit geht. Ein Plot des Diagramms wird unten gezeigt, um diese Beobachtungen zu veranschaulichen:



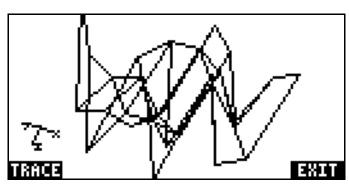
Funktion DRAW3DMATRIX

Diese Funktion nimmt als Argument eine n'm Matrix, ein **Z**, = [z_{ij}] und minimale und Maximalwerte für den Plot. Sie möchten die Werte von v_{min} und von v_{max} vorwählen, damit sie die Werte enthalten, die in **Z** verzeichnet werden. Der allgemeine Anruf zur Funktion ist folglich DRAW3DMATRIX(**Z**,v_{min},v_{max}). Um den Gebrauch von dieser Funktion zu veranschaulichen erzeugen wir zuerst eine Matrix 6'5 mit RANM({6,5}) und benennen dann Funktion DRAW3DMATRIX, wie unten gezeigt:





Der Plot ist in der Art eines FAST3DPLOT. Unterschiedliche Ansichten des Plots werden unten gezeigt:



Kapitel 13
Anwendungen der Infinitesimalrechnung

In diesem Kapitel werden Anwendungen der Taschenrechnerfunktionen auf Operationen der Infinitesimalrechnung erläutert, z. B. Grenzwerte, Ableitungen, Integrale, Potenzreihen usw.

Das Menü CALC (Infinitesimalrechnung)

Zahlreiche der in diesem Kapitel dargestellten Funktionen befinden sich im Menü CALC des Taschenrechners, das über die Tastenkombination ← CALC (der Taste 4 zugeordnet) aufgerufen wird. Das Menü CALC enthält die folgenden Einträge:

Bei den ersten vier Optionen dieses Menüs handelt es sich eigentlich um Untermenüs für (1) Ableitungen und Integrale, (2) Grenzwerte und Potenzreihen, (3) Differenzialgleichungen und (4) Grafiken. In diesem Kapitel werden die Funktionen (1) und (2) dargestellt. Differenzialgleichungen, der Inhalt von Menüpunkt (3), werden in Kapitel 16 dargestellt. Grafikfunktionen, der Inhalt von Menüpunkt (4), wurden am Ende von Kapitel 12 dargestellt. Schließlich handelt es sich bei den Menüpunkten 5. DERVX und 6. INTVX um die Funktionen, mit denen Sie eine Ableitung und ein unbestimmtes Integral für eine Funktion der CAS-Standardvariablen (normalerweise „X") erhalten. Die Funktionen DERVX und INTVX werden später ausführlicher behandelt.

Grenzwerte und Ableitungen

Bei der Differenzialrechnung werden Ableitungen (oder Änderungsraten) von Funktionen und ihren Anwendungen in der mathematischen Analyse behandelt. Die Ableitung einer Funktion ist als der Grenzwert der Differenz einer Funktion definiert, wenn das Inkrement der unabhängigen Variablen gegen Null geht. Grenzwerte werden außerdem verwendet, um die Stetigkeit von Funktionen zu überprüfen.

Funktion lim

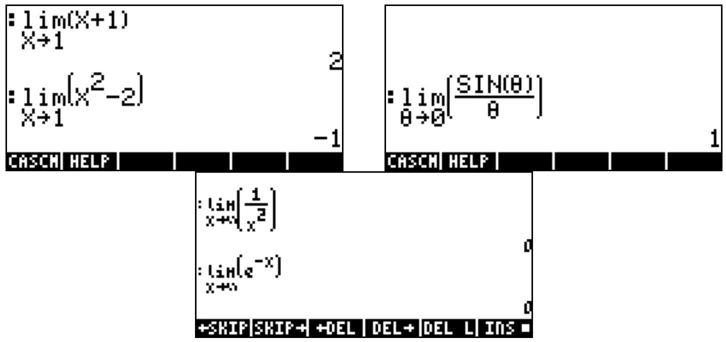
Der Taschenrechner enthält die Funktion *lim* zum Berechnen der Grenzwerte von Funktionen. Bei dieser Funktion wird ein Ausdruck als Eingangswert verwendet, der eine Funktion und ihren Wert darstellt, wobei der Grenzwert zu berechnen ist. Die Funktion *lim* kann über den Befehlskatalog (→ CAT ALPHA ← L) oder über die Option 2. LIMITS & SERIES des Menüs CALC (siehe oben) aufgerufen werden.

Hinweis: Die Funktionen im Menü LIMITS & SERIES sind unten abgebildet:

Mithilfe der Funktion DIVPC werden zwei Polynome dividiert, so dass sich eine Reihenentwicklung ergibt. Die Funktionen DIVPC, SERIES, TAYLOR0 und TAYLOR werden in Reihenentwicklungen von Funktionen verwendet und in diesem Kapitel ausführlicher erläutert.

Die Funktion *lim* wird im ALG-Modus als `lim(f(x),x=a)` eingegeben, um den Grenzwert $\lim_{x \to a} f(x)$ zu berechnen. Geben Sie im RPN-Modus zunächst die Funktion, dann den Ausdruck „x=a" und schließlich die Funktion *lim* ein. Im Folgenden werden Beispiele für den ALG-Modus sowie einige Grenzwerte gegen Unendlich dargestellt. Die Tastenkombinationen für das erste Beispiel lauten wie folgt (wenn im algebraischen Modus der Systemflag 117 auf die CHOOSE-Felder gesetzt ist):

← CALC 2 OK 2 OK X + 1 → , X → = 1 ENTER



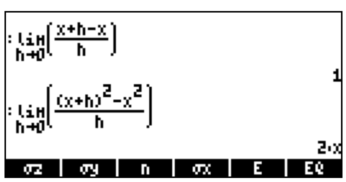
Das Unendlichkeitssymbol ist der Taste 0 zugeordnet. d. h. ., ← ∞ .

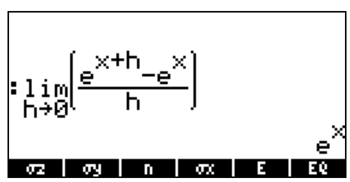
Ableitungen

Die Ableitung einer Funktion f(x) mit x = a ist definiert als der Grenzwert

$$\frac{df}{dx} = f'(x) = \lim_{h->0} \frac{f(x+h) - f(x)}{h}$$

In den folgenden Bildschirmabbildungen werden einige Beispiele für Ableitungen mit diesem Grenzwert dargestellt:

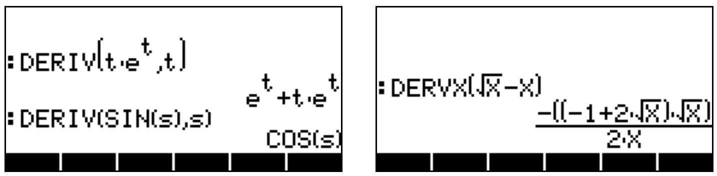




Funktionen DERIV und DERVX

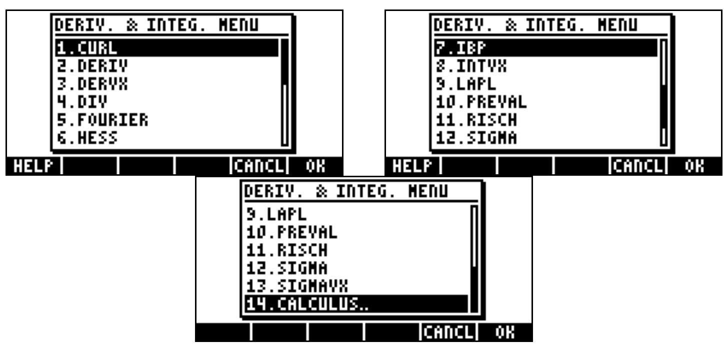
Die Funktion DERIV nimmt als Eingangswert Ableitungen von beliebigen unabhängigen Variablen an, während die Funktion DERVX Ableitungen in Bezug auf die CAS-Standardvariable VX (normalerweise „X“) als Eingangswert annimmt. Während die Funktion DERVX direkt im Menü CALC verfügbar ist, können beide Funktionen über das Menü CALCL (← CALC) im Untermenü DERIV.&INTEG aufgerufen werden.

Die Funktion DERIV erfordert eine Funktion, beispielsweise f(t), und eine unabhängige Variable, z. B. t, während für die Funktion DERVX nur eine Funktion von VX erforderlich ist. Im Folgenden werden Beispiele im ALG-Modus dargestellt. Beachten Sie, dass im RPN-Modus die Argumente vor dem Anwenden der Funktion eingegeben werden müssen.



Das Menü DERIV&INTEG

Die in diesem Untermenü verfügbaren Funktionen sind unten aufgelistet:

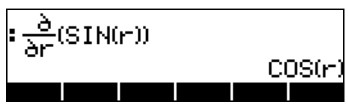


Von diesen Funktionen werden DERIV und DERVX für Ableitungen verwendet. Zu den anderen Funktionen zählen Funktionen für Stammfunktionen und Integrale (IBP, INTVX, PREVAL, RISCH, SIGMA und SIGMAVX), Fourier-Reihen (FOURIER) und Vektorrechnung (CURL, DIV, HESS, LAPL). Zunächst werden die Funktionen DERIV und DERVX erläutert. Die übrigen Funktionen werden entweder weiter unten in diesem Kapitel oder in den folgenden Kapiteln dargestellt.

Berechnen von Ableitungen mit ∂

Das Symbol ist als [→] ∂ (die [COS]-Taste) verfügbar. Dieses Symbol kann zum Eingeben einer Ableitung in den Stack oder in EquationWriter verwendet werden (siehe Kapitel 2). Wenn Sie mithilfe des Symbols eine Ableitung in den Stack eingeben, geben Sie direkt danach die unabhängige Variable und dann zwei Klammern um die abzuleitende Funktion ein. Verwenden Sie daher zum Berechnen der Ableitung d(sin(r),r) im ALG-Modus die Zeichen:
[→] ∂ [ALPHA] [←] [R] [←] () [SIN] [ALPHA] [←] [R] [ENTER]

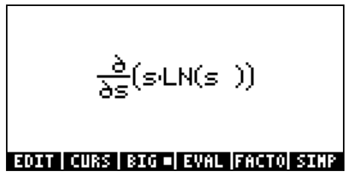
Im RPN-Modus muss dieser Ausdruck in Anführungszeichen eingeschlossen werden, bevor er in den Stack eingegeben wird. Im ALG-Modus lautet das Ergebnis:



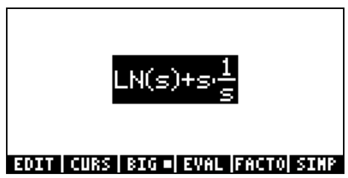
In Equation Writer gibt der Taschenrechner folgenden Ausdruck aus, wenn Sie [→] ∂, drücken:

Der Einfügecursor (◆) befindet sich rechts vom Nenner, bis Sie eine unabhängige Variable eingeben, z.B. s: [ALPHA] [←] [S]. Drücken Sie anschließend die Taste mit dem Pfeil nach rechts (▶), um den Cursor an den Platzhalter zwischen den Klammern zu verschieben.

Geben Sie anschließend die abzuleitende Funktion ein, z. B. s*ln(s):



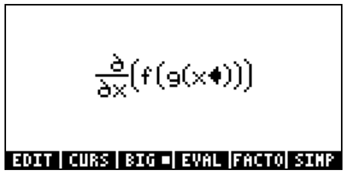
Zum Berechnen der Ableitung in EquationWriter drücken Sie viermal die Taste mit dem Pfeil nach oben, um den gesamten Ausdruck auszuwählen und drücken Sie anschließend EVAL. Die Ableitung wird in EquationWriter wie folgt berechnet:

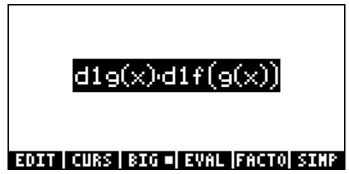


Hinweis: Das Symbol ∂ wird in der Mathematik verwendet, um eine partielle Ableitung anzugeben, d. h. die Ableitung einer Funktion mit mehreren Variablen. Jedoch werden normale und partielle Ableitungen vom Taschenrechner nicht unterschieden und für beide dasselbe Symbol verwendet. Beachten Sie diesen Unterschied, wenn Sie Ergebnisse des Taschenrechners zu Papier bringen.

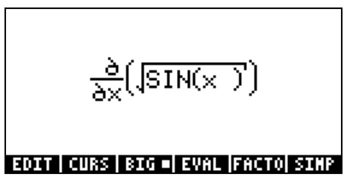
Die Kettenregel

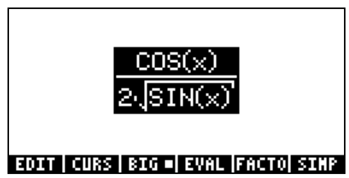
Die Kettenregel für Ableitungen wird auf Ableitungen zusammengesetzter Funktionen angewendet. Ein allgemeiner Ausdruck für die Kettenregel lautet d{f[g(x)]}/dx = (df/dg)·(dg/dx). Bei Verwendung des Taschenrechners lautet diese Formel wie folgt:





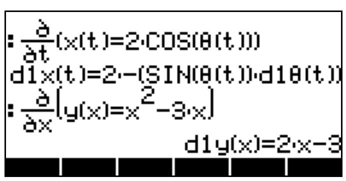
Bei dem Ausdruck d1 vor g(x) und f(g(x)) in der Formel oben handelt es sich um eine Abkürzung, die vom Taschenrechner verwendet wird, um eine erste Ableitung anzugeben, wenn die unabhängige Variable, in diesem Fall x, eindeutig definiert ist. Somit wird das Ergebnis wie in der oben dargestellten Formel für die Kettenregel interpretiert. Es folgt ein weiteres Beispiel für eine Anwendung der Kettenregel:

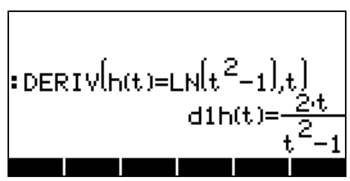


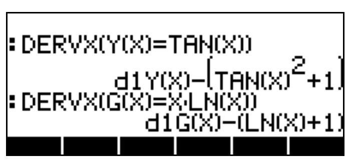


Ableitungen von Gleichungen

Mit dem Taschenrechner können Sie Ableitungen von Gleichungen berechnen, d. h. Ausdrücke, die auf beiden Seiten des Gleichheitszeichens Ableitungen enthalten. Unten werden einige Beispiele dargestellt:





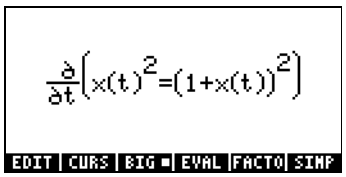


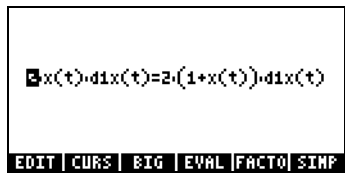
Beachten Sie, dass in den Ausdrücken, bei denen das Ableitungszeichen (∂) oder die Funktion DERIV verwendet wurde, das Gleichheitszeichen in der

Gleichung beibehalten wird, doch nicht in den Fällen, in denen die Funktion DERVX verwendet wurde. In diesen Fällen wurde die Gleichung neu geschrieben und alle zugehörigen Ausdrücke wurden auf die linke Seite des Gleichheitszeichens verschoben. Außerdem wurde das Gleichheitszeichen entfernt, doch der Ergebnisausdruck ist selbstverständlich gleich Null.

Implizite Ableitungen

Implizite Ableitungen können z. B. in folgenden Ausdrücken verwendet werden:





Anwendung von Ableitungen

Mit Ableitungen können Graphen von Funktionen berechnet und Funktionen einer Variablen optimiert werden (z. B. Suchen der Maxima und Minima). Im Folgenden werden einige Anwendungen von Ableitungen dargestellt.

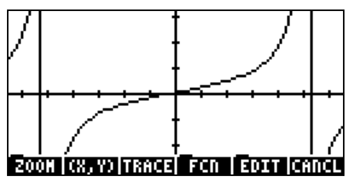
Berechnen der Graphen von Funktionen

In Kapitel 11 wurden einige Funktionen vorgestellt, die auf dem Grafikbildschirm zum Berechnen der Graphen von Funktionen der Form y = f(x) zur Verfügung stehen. Zu diesen Funktionen zählen (X,Y) und TRACE zum Bestimmen von Punkten des Graphen sowie Funktionen in den Menüs ZOOM und FCN. Mithilfe der Funktionen im Menü ZOOM können Sie die Darstellung eines Graphen vergrößern, um ihn detaillierter berechnen zu können. Diese Funktionen werden in Kapitel 12 ausführlich beschrieben. Unter den Funktionen des Menüs FCN können SLOPE, EXTR, F' und TANL zum Bestimmen der Steigung einer Tangente des Graphen, zum Bestimmen der Extremwerte (Minima und Maxima) der Funktion, zum Zeichnen der Ableitung und zum Bestimmen der Gleichung für die Tangente verwendet werden.

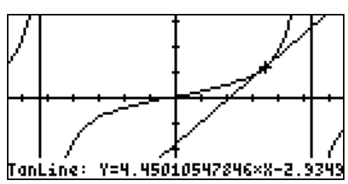
Probieren Sie folgendes Beispiel für die Funktion y = tan(x) aus.

- Drücken Sie im RPN-Modus gleichzeitig ⌐ 2D/3D ,, um das Fenster PLOT SETUP zu öffnen.
- Ändern Sie ggf. mithilfe von [CHOOS] TYPE in FUNCTION.
- Drücken Sie ▼ und geben Sie die Gleichung „TAN(X)" ein.
- Stellen Sie sicher, dass die unabhängige Variable auf „X" gesetzt ist.
- Drücken Sie NXT OK, um zur normalen Anzeige des Taschenrechners zurückzukehren.
- Drücken Sie gleichzeitig ⌐ WIN , um das PLOT-Fenster aufzurufen.
- Ändern Sie den H-VIEW-Bereich auf -2 bis 2 und den V-VIEW-Bereich auf -5 bis 5.
- Drücken Sie ERASE DRAW, um die Funktion in Polarkoordinaten zu zeichnen.

Die resultierende Zeichnung wird wie folgt dargestellt:



- Beachten Sie die vertikalen Linien, die die Asymptoten darstellen. Diese sind nicht Bestandteil des Graphen, sondern zeigen die Punkte an, an denen TAN(X) bei bestimmten Werten von X gegen ± ∞ geht.
- Drücken Sie TRACE (X,Y) und bewegen Sie den Cursor an Punkt X: 1.08E0, Y: 1.86E0. Drücken Sie anschließend NXT FCN SLOPE. Das Ergebnis ist die Steigung: 4.45010547846.
- Drücken Sie NXT NXT TANL. Hierdurch wird die Gleichung der Tangente erstellt und der zugehörige Graph in dieselbe grafische Darstellung gezeichnet. Das Ergebnis wird in der Abbildung unten dargestellt:

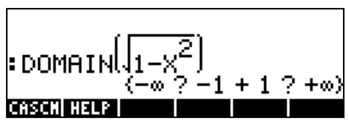


- Drücken Sie NXT PICT CANCL ON ,um zur normalen Anzeige des Taschenrechners zurückzukehren. Beachten Sie, dass die gewünschte Steigung und Tangente im Stack aufgelistet sind.

Funktion DOMAIN

Mit der über den Befehlskatalog (→ _CAT) verfügbaren Funktion DOMAIN erhalten Sie den Definitionsbereich einer Funktion als eine Liste von Zahlen und Beschreibungen. Beispielsweise gibt

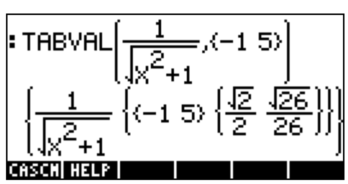
an, dass die Funktion LN(X) zwischen –∞ und 0 nicht definiert ist (?), dass sie jedoch zwischen 0 und +∞ definiert ist (+). Andererseits gibt



an, dass die Funktion weder zwischen –∞ und -1 noch zwischen 1 und +∞ definiert ist. Der Definitionsbereich dieser Funktion lautet daher -1<X<1.

Funktion TABVAL

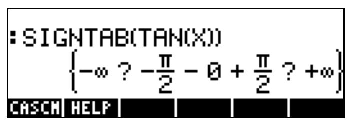
Diese Funktion wird über den Befehlskatalog oder im Menü CALC über das Untermenü GRAPH aufgerufen. Als Argument der Funktion TABVAL ist die Funktion der CAS-Variablen f(X) und eine Liste zweier Zahlen erforderlich, die den relevanten Bereich für die Funktion f(X) darstellen. Die Funktion TABVAL gibt die Eingabewerte plus den Bereich der Funktion zurück, der dem für die Eingabe verwendeten Bereich entspricht. Beispiel:



Durch dieses Ergebnis wird angegeben, dass der Bereich der Funktion $f(X) = \dfrac{1}{\sqrt{X^2+1}}$, der dem Definitionsbereich D = { -1,5 } entspricht, B = $\left\{ \dfrac{\sqrt{2}}{2}, \dfrac{\sqrt{26}}{26} \right\}$ ist.

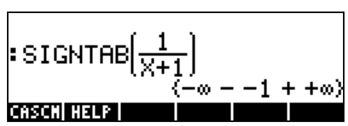
Funktion SIGNTAB

Mit der über den Befehlskatalog (↱ CAT) aufrufbaren Funktion SIGNTAB erhalten Sie Informationen über das Vorzeichen einer Funktion in ihrem Definitionsbereich. Beispielsweise gibt SIGNTAB für die Funktion TAN(X)



an, dass TAN(X) zwischen –π/2 und 0 negativ und zwischen 0 und π/2 positiv ist. In diesem Fall liefert SIGNTAB in den Intervallen zwischen –∞ und -π/2 und zwischen +π/2 und ∞ keine Informationen (?). Somit liefert SIGNTAB in diesem Fall ausschließlich Informationen über den Hauptdefinitionsbereich von TAN(X), also -π/2 < X < +π/2.

Unten ist ein zweites Beispiel für die Funktion SIGNTAB abgebildet:



In diesem Fall ist die Funktion für X<-1 negativ und für X>-1 positiv.

Funktion TABVAR

Diese Funktion wird über den Befehlskatalog oder im Menü CALC über das Untermenü GRAPH aufgerufen. Als Eingangswert wird die Funktion f(VX) verwendet, wobei VX die CAS-Standardvariable ist. Die Funktion gibt im RPN-Modus Folgendes zurück:

- Ebene 3: die Funktion f(VX).

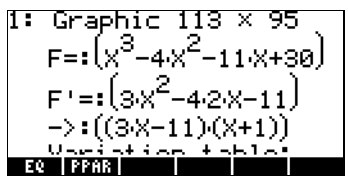
- Zwei Listen. Die erste Liste gibt die Abweichung der Funktion (d. h. die Punkte, an denen die Werte ab- oder zunehmen) in Bezug auf die unabhängige Variable VX an, die zweite Liste die Abweichung der Funktion in Bezug auf die abhängige Variable.

- Ein Grafikobjekt, das anzeigt, wie die Abweichungstabelle berechnet wurde.

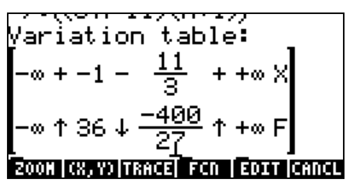
Beispiel: Berechnen Sie mithilfe der Funktion TABVAR die Funktion Y = X^3-4X^2-11X+30. Verwenden Sie im RPN-Modus die folgenden Tastenkombinationen:

`'X^3-4*X^2-11*X+30'` ENTER → _CAT ALPHA T (Auswahl von TABVAR) OK

Der Taschenrechner zeigt auf Ebene 1 des Stacks Folgendes an:



Dies ist ein Grafikobjekt. Um das gesamte Ergebnis anzeigen zu können, drücken Sie ▽. Die Abweichungstabelle der Funktion wird wie folgt dargestellt:



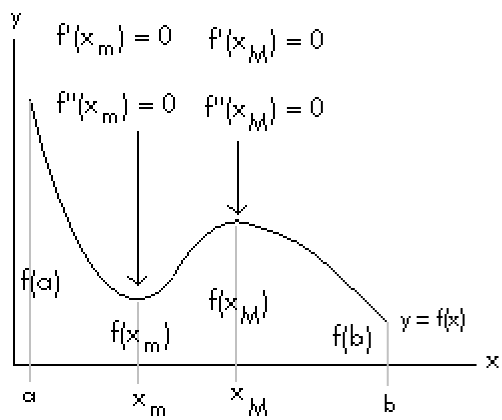
Drücken Sie ON, um zur normalen Anzeige des Taschenrechners zurückzukehren. Drücken Sie ◄, um dieses letzte Ergebnis aus dem Stack zu entfernen.

Ebene 1 enthält nun zwei Listen, die der obersten und untersten Zeile der zuvor dargestellten Grafikmatrix entsprechen. Diese Listen können zum Programmieren verwendet werden. Drücken Sie ◄, um dieses letzte Ergebnis aus dem Stack zu entfernen.

Im Folgenden wird die oben dargestellte Abweichungstabelle erläutert: Die Funktion F(X) nimmt für X im Intervall (-∞, -1) zu und erreicht das Maximum 36 bei X = -1. Dann nimmt f(X) bis X = 11/3 ab und erreicht das Minimum -400/27. Anschließend nimmt F(X) bis +∞ zu. Außerdem ist bei X = ±∞ auch F(X) = ±∞.

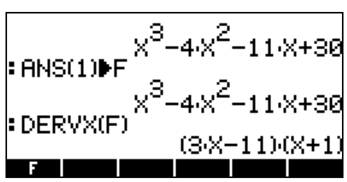
Verwenden von Ableitungen zum Berechnen von Extrempunkten

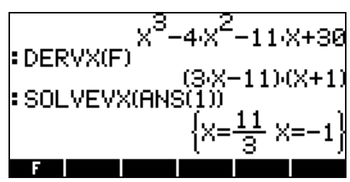
„Extrempunkte" bzw. Extremwerte sind die allgemeine Bezeichnung für die Maximal- und Minimalwerte einer Funktion in einem bestimmten Intervall. Da die Ableitung einer Funktion an einem bestimmten Punkt die Steigung der Tangente zur Kurve an diesem Punkt darstellt, stellen die Werte von x für f'(x) = 0 die Punkte dar, an denen der Graph der Funktion das Maximum oder Minimum erreicht. Darüber hinaus gibt der Wert der zweiten Ableitung der Funktion f"(x) an diesen Punkten an, ob der Punkt ein *relatives oder lokales Maximum* [f"x)<0] bzw. *Minimum* [f"(x)>0] ist. Dies wird in der folgenden Abbildung veranschaulicht.



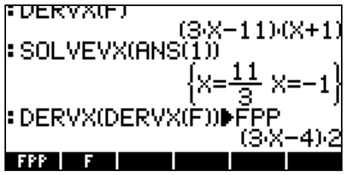
In dieser Abbildung beschränken wir uns darauf, die Extrempunkte der Funktion y = f(x) im x-Intervall [a,b] zu bestimmen. In diesem Intervall befinden sich zwei Punkte, $x = x_m$ und $x = x_M$, wobei f'(x)=0 ist. Der Punkt $x = x_m$, wobei f"(x)>0 ist, stellt ein lokales Minimum dar, wenn der Punkt $x = x_M$, wobei f"(x)<0 ist, stellt ein lokales Maximum dar. Aus dem Graph von y = f(x) folgt, dass sich das absolute Maximum im Intervall [a,b] bei x = a und das absolute Minimum bei x = b befindet.

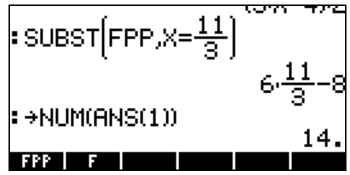
Um beispielsweise zu bestimmen, wo sich die kritischen Punkte der Funktion `X^3-4*X^2-11*X+30 befinden, können wir im ALG-Modus die folgenden Eingaben verwenden:



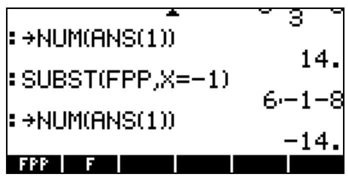


Wir ermitteln zwei kritische Punkte, bei x = 11/3 und bei x = -1. Geben Sie zum Berechnen der zweiten Ableitung an jedem Punkt Folgendes ein:

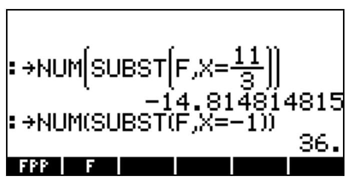




Auf dem letzten Bildschirm wird angezeigt, dass f″(11/3) = 14, so dass x = 11/3 ein relatives Minimum ist. Für x = -1 gilt Folgendes:



Dieses Ergebnis bedeutet, dass f″(-1) = -14, so dass x = -1 ein relatives Maximum ist. Berechnen Sie die Funktion an diesen Punkten, um zu überprüfen, ob tatsächlich gilt: (-1) > f(11/3).



Ableitungen höherer Ordnung

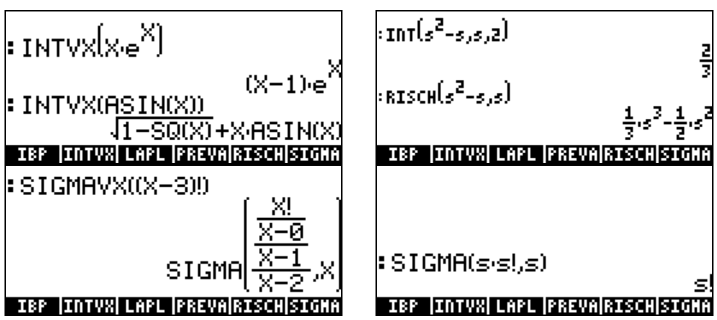
Ableitungen höherer Ordnung können durch mehrfaches Anwenden einer Ableitungsfunktion berechnet werden. Beispiel:

Stammfunktionen und Integrale

Die Stammfunktion einer Funktion f(x) ist die Funktion F(x), wobei f(x) = dF/dx. Da z. B. $d(x^3)/dx = 3x^2$, lautet für $f(x) = 3x^2$ die Stammfunktion $F(x) = x^3 + C$, wobei C eine Konstante ist. Eine Stammfunktion kann als <u>unbestimmtes Integral</u> dargestellt werden, d. h. $\int f(x)dx = F(x) + C$, wenn und nur wenn f(x) = dF/dx und C eine Konstante ist.

Funktionen INT, INTVX, RISCH, SIGMA und SIGMAVX

Der Taschenrechner enthält zum Berechnen der Stammfunktionen von Funktionen die Funktionen INT, INTVX, RISCH, SIGMA und SIGMAVX. Die Funktionen INT, RISCH und SIGMA können mit Funktionen beliebiger Variablen verwendet werden, während die Funktionen INTVX und SIGMAVX die Funktionen der CAS-Variablen VX (normalerweise „X“) verwenden. Die Funktionen INT und RISCH erfordern daher nicht nur den Ausdruck für die zu integrierende Funktion, sondern auch den Namen der unabhängigen Variablen. Die Funktion INT erfordert außerdem einen Wert von x, wenn die Stammfunktion berechnet wird. Die Funktionen INTVX und SIGMAVX erfordern nur den Ausdruck der in Bezug auf VX zu integrierenden Funktion. Im Folgenden werden einige Beispiele im ALG-Modus dargestellt.



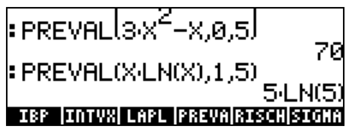
Beachten Sie, dass die Funktionen SIGMAVX und SIGMA für Integranden konzipiert sind, die eine ganzzahlige Funktion, z. B. die oben dargestellte Fakultätsfunktion (!), umfassen. Sie ergeben eine so genannte diskrete Ableitung, d. h. eine ausschließlich für ganze Zahlen definierte Ableitung.

Bestimmte Integrale

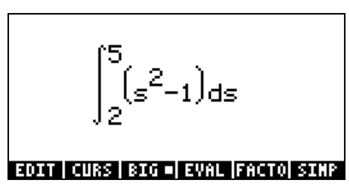
Bei einem bestimmten Integral einer Funktion wird die resultierende Stammfunktion am oberen und unteren Grenzwert eines Intervalls (a,b) berechnet und die berechneten Werte werden subtrahiert. Die Formel lautet

$\int_a^b f(x)dx = F(b) - F(a),$ wobei f(x) = dF/dx.

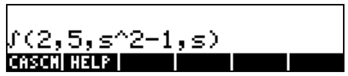
Verwenden Sie die Funktion PREVAL(f(x),a,b), um bestimmte Integrale mit der CAS-Variablen VX (normalerweise „X") zu berechnen. Beispiel:



Der Taschenrechner verfügt zum Berechnen bestimmter Integrale auch über das Integralsymbol als Tastenkombination ⇗ ∫ (der TAN-Taste zugeordnet). Integrale können am einfachsten in EquationWriter erstellt werden (ein Beispiel hierfür finden Sie in Kapitel 2). In EquationWriter erhalten Sie mit dem Symbol ⇗ ∫ das Integralzeichen sowie Platzhalter für die Integrationsgrenzen (a,b), für die Funktion f(x) und für die Variable (x) der Integration. In den folgenden Bildschirmabbildungen wird das Erstellen eines bestimmten Integrals veranschaulicht. Der Einfügecursor befindet sich zunächst an der unteren Grenze der Integration. Geben Sie einen Wert ein und drücken Sie die Taste mit dem Pfeil nach rechts (▶), um zur oberen Grenze der Integration zu wechseln. Geben Sie an dieser Position einen Wert ein und drücken Sie erneut ▶, um zur Position des Integranden zu wechseln. Geben Sie den Ausdruck für den Integranden ein und drücken Sie die Taste erneut, um zum Platzhalter des Differenzials zu wechseln, geben Sie an dieser Position die Variable der Integration ein, und das Integral kann berechnet werden.



Nun können Sie ENTER drücken, um das Integral an den Stack zurückzugeben, wie in der folgenden Eingabe (ALG-Modus) dargestellt:



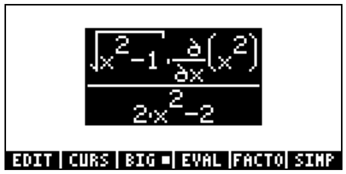
Dies ist das allgemeine Format für das bestimmte Integral, wenn es direkt in den Stack eingegeben wird, d. h. ∫(untere Grenze, obere Grenze, Integrand, Variable der Integration).

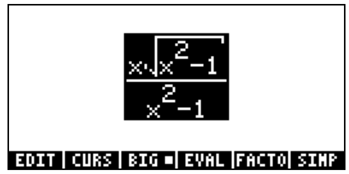
Wenn Sie an dieser Stelle ENTER drücken, wird das Integral im Stack berechnet:

Das Integral kann auch in EquationWriter berechnet werden, indem Sie den gesamten Ausdruck auswählen und die Menütaste EVAL verwenden.

Schrittweise Berechnung von Ableitungen und Integralen

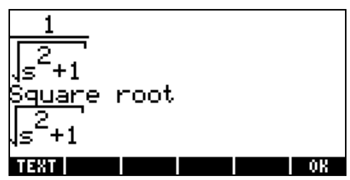
Wenn in den CAS MODES-Fenstern die Option Step/Step ausgewählt ist (siehe Kapitel 1), wird die Berechnung von Ableitungen und Integralen in einzelnen Schritten angezeigt. In der folgenden Abbildung wird z. B. die Berechnung einer Ableitung in EquationWriter dargestellt:

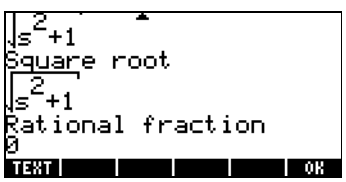


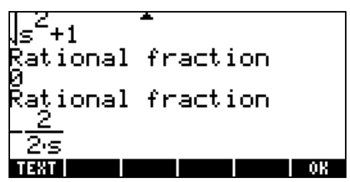


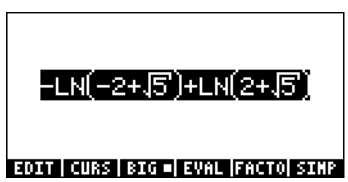
Beachten Sie die Anwendung der Kettenregel im ersten Schritt, wobei die Ableitung der Funktion unter dem Integral explizit im Zähler bleibt. Im zweiten Schritt wird der resultierende Bruch in eine rationale Zahl umgewandelt (die Quadratwurzel aus dem Nenner entfernt) und vereinfacht. Im dritten Schritt wird die endgültige Version angezeigt. Jeder Schritt wird durch Drücken der Menütaste EVAL angezeigt, bis der Punkt erreicht ist, an dem der Ausdruck durch die weitere Anwendung der Funktion EVAL nicht mehr geändert wird.

Im folgenden Beispiel wird die schrittweise Berechnung eines bestimmten Integrals in EquationWriter dargestellt:





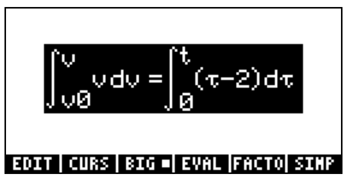


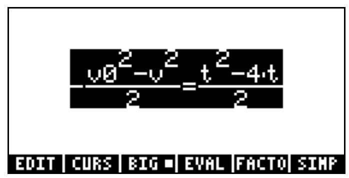


Beachten Sie, dass durch den schrittweisen Vorgang Informationen über die von CAS zum Lösen dieses Integrals ausgeführten Zwischenschritte bereitgestellt werden. CAS bestimmt zunächst ein Quadratwurzelintegral, dann einen rationalen Bruch sowie einen zweiten rationalen Ausdruck und liefert dann das Endergebnis. Beachten Sie, dass diese Schritte für den Taschenrechner von großer Bedeutung sind, obwohl für den Benutzer nicht genügend Informationen über die einzelnen Schritte geboten werden.

Integrieren einer Gleichung

Das Integrieren eine Gleichung ist unkompliziert. Der Taschenrechner integriert einfach beide Seiten der Gleichung gleichzeitig, z. B.



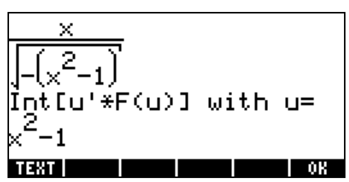


Methofden der Integration

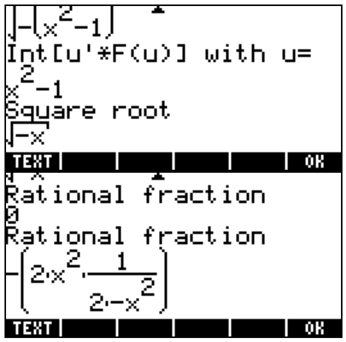
Wie in den folgenden Beispielen gezeigt, können mit dem Taschenrechner mehrere Integrationsmethoden angewendet werden.

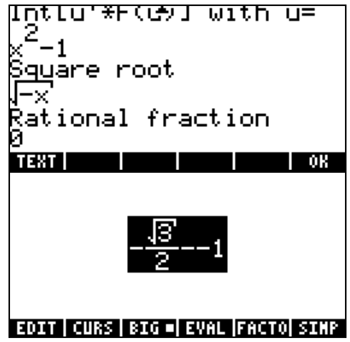
Substitution oder Ändern von Variablen

Angenommen, wir möchten das Integral $\int_0^2 \frac{x}{\sqrt{1-x^2}}dx$ berechnen. Bei einer schrittweisen Berechnung in EquationWriter lautet die Abfolge der Variablensubstitutionen wie folgt:



Im zweiten Schritt wird die zu verwendende ordnungsgemäße Substitution dargestellt, u = x^2-1.





In den letzten vier Schritten wird die Entwicklung der Lösung gezeigt: eine Quadratwurzel, dann ein Bruch, ein zweiter Bruch und das Endergebnis. Das Ergebnis kann unter Verwendung der Funktion SIMP vereinfacht werden und wird wie folgt angezeigt:

Partielle Integration und Differenziale

Ein Differenzial einer Funktion y = f(x) ist als dy = f'(x) dx definiert, wobei f'(x) die Ableitung von f(x) ist. Differenziale werden für die Darstellung kleiner Inkremente in den Variablen verwendet. Das Differenzial eines Produkts zweier Funktionen y = u(x)v(x) wird durch dy = u(x)dv(x) + du(x)v(x) oder einfach durch d(uv) = udv - vdu berechnet. Somit wird das Integral von udv = d(uv) - vdu als $\int udv = \int d(uv) - \int vdu$ geschrieben. Da gemäß der Definition eines Differenzials ∫dy = y ist, schreiben wir den vorherigen Ausdruck als

$$\int udv = uv - \int vdu \; .$$

Diese als partielle Integration bezeichnete Formel kann zum Bestimmen eines Integrals verwendet werden, wenn dv auf einfache Weise integriert werden kann. Beispielsweise kann das Integral $\int xe^x dx$ durch partielle Integration bestimmt werden, wenn wir u = x, dv = e^xdx verwenden, da v = e^x. Wenn du = dx ist, lautet das Integral $\int xe^x dx = \int u dv = uv - \int v du = xe^x - \int e^x dx = xe^x - e^x$.

Der Taschenrechner enthält im Menü CALC/DERIV&INTG die Funktion IBP, die als Argumente die ursprüngliche zu integrierende Funktion, also u(X)*v'(X), und die Funktion v(X) annimmt, und gibt u(X)*v(X) sowie -v(X)*u'(X) zurück. Mit anderen Worten, die Funktion IBP gibt die beiden Ausdrücke auf der rechten Seite der Gleichung der partiellen Integration zurück. Für das oben verwendete Beispiel können wir im ALG-Modus Folgendes schreiben:

Somit können wir die Funktion IBP verwenden, um die Komponenten einer partiellen Integration bereitzustellen. Der nächste Schritt muss separat ausgeführt werden.

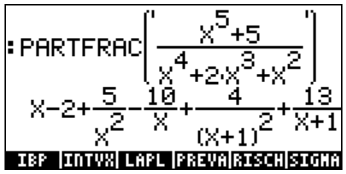
Es ist wichtig zu erwähnen, dass das Integral direkt berechnet werden kann, indem z. B. folgende Eingabe verwendet wird:

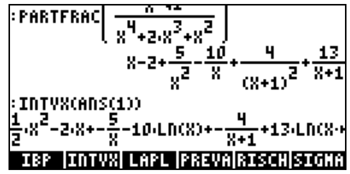
Integration durch Partialbruchzerlegung

Die in Kapitel 5 vorgestellte Funktion PARTFRAC ermöglicht die Zerlegung eines Bruches in Partialbrüche. Diese Methode empfiehlt sich, um einen komplizierten Bruch in eine Summe einfacher Brüche zu zerlegen, die dann nacheinander integriert werden können. Um beispielsweise

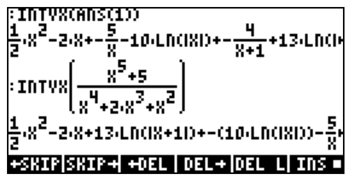
$$\int \frac{X^5 + 5}{X^4 + 2X^3 + X} dX$$

zu integrieren, können wir den Bruch wie folgt in seine Partialbrüche zerlegen:





Die direkte Integration führt zum gleichen Ergebnis, wobei einige Ausdrücke umgestellt werden (in CAS festgelegter rigoroser Modus, siehe Kapitel 2):

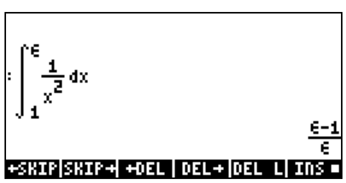


Unzulässige Integrale

Hierbei handelt es sich um Integrale mit Unendlich als Integrationsgrenze. Üblicherweise wird bei einem unzulässigen Integral zunächst das Integral als Grenzwert gegen Unendlich berechnet, z. B.

$$\int_1^\infty \frac{dx}{x^2} = \lim_{\varepsilon \to \infty} \int_1^\varepsilon \frac{dx}{x^2}.$$

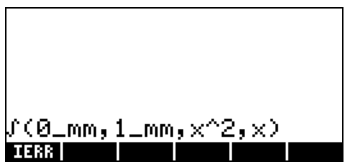
Mit dem Taschenrechner setzen wir die Berechnung wie folgt fort:

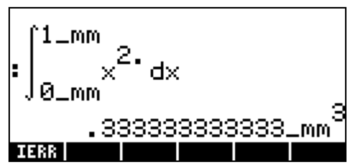


Stattdessen können Sie auch das Integral mit Unendlich als Grenzwert sofort berechnen, z. B.:

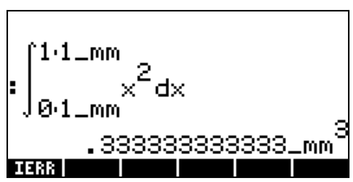
Integralrechnungen mit Einheiten

Eine Integrale kann auch mit, innerhalb der Grenzwerte der Integralen eingeschlossenen Einheiten, berechnet werden, wie man das im nachfolgenden Beispiel im ALG-Modus, mit CAS auf Approx Modus eingestellt, sehen kann. Auf der linken Seite ist die im Editor eingetippte Integrale vor drücken der Taste ENTER zu sehen. Auf der rechten Seite das Ergebnis, nach drücken der Taste ENTER.





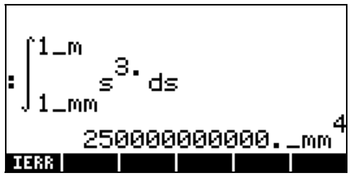
Geben Sie die Integrale mit dem CAS auf Exact Modus eingestellt ein, erhalten Sie die Meldung in den Approx Modus umzustellen, die Grenzen der Integrale aber, werden in einem anderen, als hier gezeigten, Format ausgegeben:

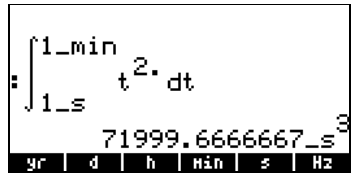


Die hier gezeigten Grenzen sind 1×1_mm und 0×1_mm, welches 1_mm und 0_mm, wie vorher gezeigt entspricht. Beachten Sie nur die unterschiedlichen Ausgabeformate.

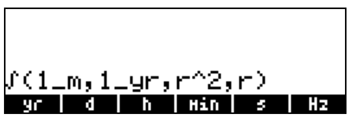
Einige Notizen bei der Verwendung von Einheiten innerhalb der Grenzwerte der Integrale:

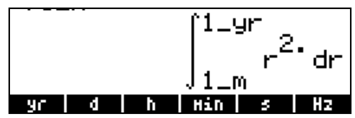
1 – Die Einheiten der unteren Grenze der Integrale werden die im Endresultat gezeigten sein, wie das im nachstehenden Beispiel veranschaulicht wird:



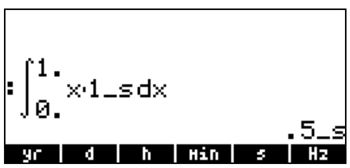


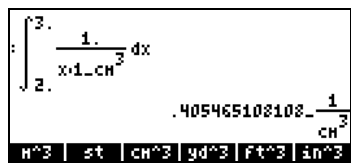
2 – Die oberen Grenzwerte müssen mit den unteren konsistent sein. Andernfalls ist das Ergebnis eine nicht ausgewertete Integrale. So zum Beispiel:



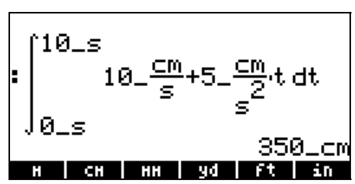


3 – Der Integrand könnte zwei Einheiten beinhalten. So zum Beispiel:





4 – Beinhalten beide, die Grenzwerte wie auch der Integrand, Einheiten, wird das Ergebnis eine Kombination, entsprechend den Regeln der Integrale sein, dieser Einheiten darstellen. So zum Beispiel:



Unendliche Reihen

Eine unendliche Reihe hat die Form $\sum_{n=0,1}^{\infty} h(n)(x-a)^n$. Unendliche Reihen beginnen normalerweise mit dem Index n = 0 oder n = 1. Jedes Glied der Reihe besitzt einen Koeffizienten h(n), der vom Index n abhängt.

Taylor- und MacLaurin-Reihen

Eine Funktion f(x) kann mit einer Taylor-Reihe zu einer unendlichen Reihe um einen Punkt $x=x_0$ entwickelt werden, also

$$f(x) = \sum_{n=0}^{\infty} \frac{f^{(n)}(x_o)}{n!} \cdot (x - x_o)^n \ ,$$

wobei $f^{(n)}(x)$ die n-te Ableitung von f(x) darstellt, mit x, $f^{(0)}(x) = f(x)$.

Wenn x_0 gleich Null ist, wird die Reihe als MacLaurin-Reihe bezeichnet, d. h.

$$f(x) = \sum_{n=0}^{\infty} \frac{f^{(n)}(0)}{n!} \cdot x^n$$

Taylor-Polynom und Rest

In der Realität können nicht alle Glieder einer unendlichen Reihe berechnet werden. Stattdessen berechnen wir mit einem Polynom der Ordnung k, $P_k(x)$ einen Näherungswert für die Reihe und schätzen die Ordnung eines Residuums $R_k(x)$, so dass

$$f(x) = \sum_{n=0}^{k} \frac{f^{(n)}(x_o)}{n!} \cdot (x - x_o)^n + \sum_{n=k+1}^{\infty} \frac{f^{(n)}(x_o)}{n!} \cdot (x - x_o)^n \ ,$$

d. h. $$f(x) = P_k(x) + R_k(x).$$

Das Polynom $P_k(x)$ wird als Taylor-Polynom bezeichnet. Die Ordnung des Residuums wird als eine kleine Menge $h = x\text{-}x_0$ geschätzt, d. h., das Polynom wird bei einem Wert von x berechnet, der sehr nah an x_0 liegt. Das Residuum wird durch

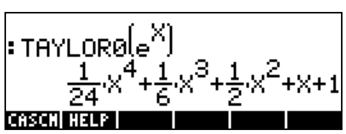
$$R_k(x) = \frac{f^{(k+1)}(\xi)}{k!} \cdot h^{k+1},$$

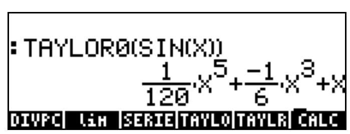
angegeben, wobei ξ eine Zahl nahe $x = x_0$ ist. Da ξ im Gegensatz zu einem Schätzwert für das Residuum normalerweise nicht bekannt ist, geben wir einen Schätzwert für die Ordnung des Residuums in Bezug auf h an, d. h., wir sagen, dass $R_k(x)$ einen Fehler der Ordnung h^{n+1} aufweist, oder $R \approx O(h^{k+1})$. Wenn h eine kleine Zahl ist, z. B. $h<<1$, ist h^{k+1} in der Regel sehr klein, d. h. $h^{k+1}<<h^k<< \ldots << h << 1$. Je größer daher für x nahe x_0 die Zahl der Elemente im Taylor-Polynom ist, desto niedriger ist die Ordnung des Residuums.

Funktionen TAYLR, TAYLR0 und SERIES

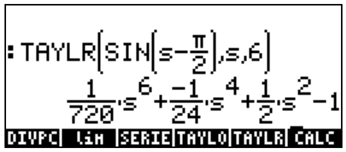
Die Funktionen TAYLR, TAYLR0 und SERIES werden zum Erzeugen von Taylor-Polynomen sowie für Taylor-Reihen mit Residuen verwendet. Diese Funktionen sind im Menü CALC/LIMITS&SERIES verfügbar, das bereits in diesem Kapitel beschrieben wurde.

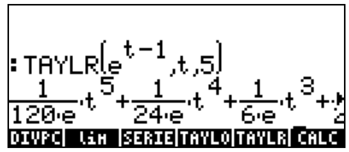
Die Funktion TAYLOR0 ergibt eine MacLaurin-Entwicklung (um X = 0) eines Ausdrucks in der unabhängigen Standardvariablen VX (in der Regel „X"). Bei der Entwicklung wird eine relative Potenz 4. Ordnung verwendet, d. h. die Differenz zwischen der höchsten und niedrigsten Potenz beträgt 4. Beispiel:





Die Funktion TAYLR ergibt die Taylor-Entwicklung der Funktion einer beliebigen Variablen x um einen Punkt x = a für die vom Benutzer angegebene Ordnung k. Somit hat die Funktion das Format TAYLR(f(x-a),x,k). Beispiel:





Die Funktion SERIES ergibt ein Taylor-Polynom, wobei als Argumente die zu entwickelnde Funktion f(x), nur ein Variablenname (bei MacLaurin-Reihen) oder ein Ausdruck der Form „Variable = Wert", der den Entwicklungspunkt der Taylor-Reihe angibt, und die Ordnung der zu erzeugenden Reihe verwendet werden. Die Funktion SERIES gibt zwei Ausgabeobjekte zurück: Eine Liste mit vier Elementen und ein Ausdruck für h = x - a, wenn das zweite Argument im Funktionsaufruf „x = a" lautet, also ein Ausdruck für das Inkrement h ist. Die als erstes Ausgabeobjekt zurückgegebene Liste enthält die folgenden Elemente:

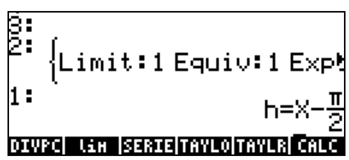
1 -Einen bidirektionalen Grenzwert der Funktion am Entwicklungspunkt, d. h. $\lim_{x \to a} f(x)$

2 - Einen Äquivalenzwert der Funktion nahe x = a

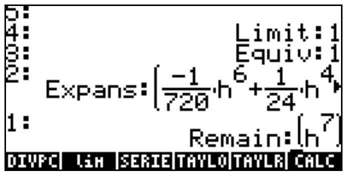
3 - Einen Ausdruck für das Taylor-Polynom

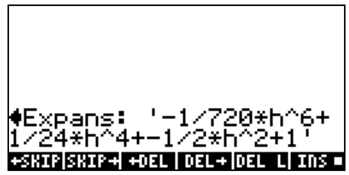
4 - Die Ordnung des Residuums bzw. Restes

Wegen der relativ umfangreichen Ausgabe ist diese Funktion im RPN-Modus leichter zu handhaben. Beispiel:



Entfernen Sie den Inhalt der Ebene 1 des Stacks, indem Sie ◄ drücken und geben sie dann EVAL ein, um die Liste in ihre Bestandteile zu zerlegen. Die Ergebnisse lauten wie folgt:





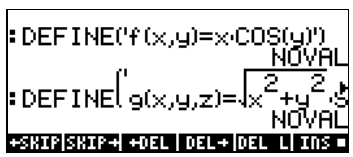
In der Abbildung rechts oben wird der Befehlszeileneditor verwendet, um die Reihenentwicklung im Detail anzuzeigen.

Kapitel 14
Anwendungen multivariater Infinitesimalrechnung

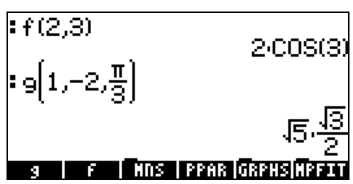
Die Bezeichnung „Multivariate Infinitesimalrechnung“ bezieht sich auf Funktionen mit mindestens zwei Variablen. In diesem Kapitel werden die Grundkonzepte der multivariaten Infinitesimalrechnung einschließlich partieller Ableitungen und mehrfacher Integrale erläutert:

Multivariate Funktionen

Eine Funktion mit mindestens zwei Variablen kann im Taschenrechner mit der Funktion DEFINE (← DEF) definiert werden. Um das Konzept partieller Ableitungen zu veranschaulichen, definieren wir zwei multivariate Funktionen f(x,y) = x cos(y) und g(x,y,z) = $(x^2+y^2)^{1/2}$sin(z) wie folgt:



Wir können diese Funktionen wie jede andere Taschenrechnerfunktion berechnen, z. B.



Graphen zweidimensionaler Funktionen können mit Fast3D-, Wireframe-, Ps-Contour-, Y-Slice-, Gridmap- und Pr-Surface-Zeichnungen erstellt werden, wie in Kapitel 12 beschrieben.

Partielle Ableitungen

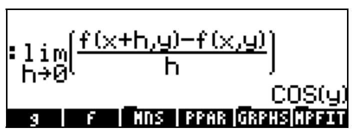
Betrachten Sie die Funktion mit zwei Variablen z = f(x,y). Die partielle Ableitung der Funktion für x ist definiert durch den Grenzwert

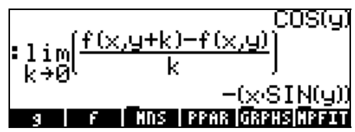
$$\frac{\partial f}{\partial x} = \lim_{h \to 0} \frac{f(x+h,y) - f(x,y)}{h} \quad .$$

Entsprechend ist

$$\frac{\partial f}{\partial y} = \lim_{k \to 0} \frac{f(x,y+k) - f(x,y)}{k} .$$

Wir verwenden die zuvor definierten multivariaten Funktionen, um mit diesen Definitionen partielle Ableitungen zu berechnen. Dies sind die Ableitungen von f(x,y) für x bzw. y:





Beachten Sie, dass die Definition der partiellen Ableitung für x z. B. erfordert, dass y unverändert bleibt, während als Grenzwert h→0 verwendet wird. Dies bietet eine Möglichkeit zur schnellen Berechnung partieller Ableitungen multivariater Funktionen: Verwenden Sie die Regeln gewöhnlicher Ableitungen für die relevante Variable, während alle anderen Variablen als Konstanten betrachtet werden. So ist z. B.

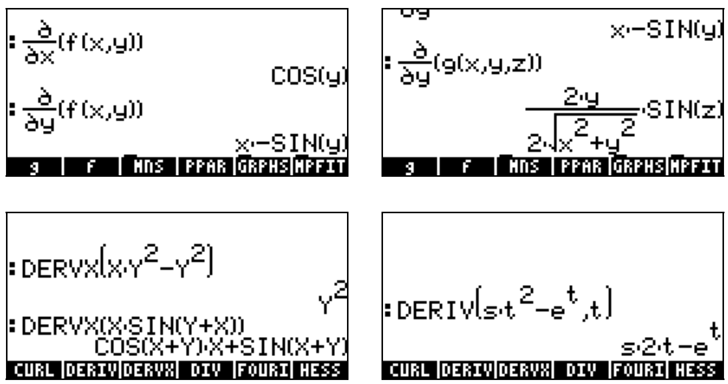
$$\frac{\partial}{\partial x}\left(x\cos(y)\right) = \cos(y), \ \frac{\partial}{\partial y}\left(x\cos(y)\right) = -x\sin(y) ,$$

wobei es sich um dieselben Ergebnisse wie bei den zuvor berechneten Grenzwerten handelt.Betrachten Sie ein weiteres Beispiel:

$$\frac{\partial}{\partial x}\left(yx^2 + y^2\right) = 2yx + 0 = 2xy$$

Bei dieser Berechnung behandeln wir y als Konstante und berechnen Ableitungen des Ausdrucks für x.

Entsprechend können Sie die Ableitungsfunktionen des Taschenrechners, z. B. DERVX, DERIV und ∂ (in Kapitel 13 ausführlich beschrieben), zum Berechnen von partiellen Ableitungen verwenden. Entsinnen Sie sich, dass die Funktion DERVX die CAS-Standardvariabel VX (in der Regel „X“) verwendet. Sie können daher mit DERVX nur Ableitungen für X berechnen. Im Folgenden werden einige Beispiele für partielle Ableitungen erster Ordnung dargestellt:



Ableitungen höherer Ordnung

Die folgenden Ableitungen höherer Ordnung können als

$$\frac{\partial^2 f}{\partial x^2} = \frac{\partial}{\partial x}\left(\frac{\partial f}{\partial x}\right), \frac{\partial^2 f}{\partial y^2} = \frac{\partial}{\partial y}\left(\frac{\partial f}{\partial y}\right),$$

$$\frac{\partial^2 f}{\partial y \partial x} = \frac{\partial}{\partial y}\left(\frac{\partial f}{\partial x}\right), \frac{\partial^2 f}{\partial x \partial y} = \frac{\partial}{\partial x}\left(\frac{\partial f}{\partial y}\right)$$

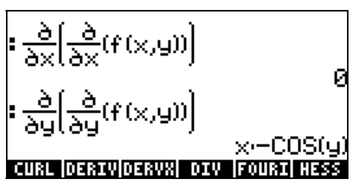
definiert werden. Die letzten beiden Ausdrücke stellen gemischte Ableitungen dar. Die Zeichen im Nenner für partielle Ableitungen geben die Ordnung der

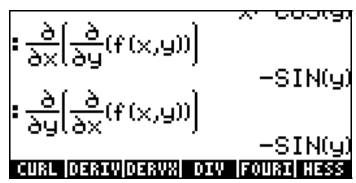
Ableitung an. Auf der linken Seite wird die Ableitung zunächst für x und dann für y berechnet, und auf der rechten Seite ist die Reihenfolge umgekehrt. Es ist wichtig darauf hinzuweisen, dass bei einer stetigen und differenzierbaren Funktion Folgendes gilt:

$$\frac{\partial^2 f}{\partial y \partial x} = \frac{\partial^2 f}{\partial x \partial y}.$$

Ableitungen dritter, vierter und höherer Ordnung werden auf ähnliche Weise definiert.

Um mit dem Taschenrechner Ableitungen höherer Ordnung zu berechnen, wenden Sie einfach die Ableitungsfunktion so häufig wie erforderlich an. Unten werden einige Beispiele dargestellt:



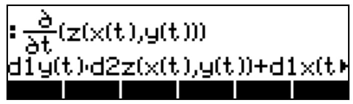


Die Kettenregel für partielle Ableitungen

Betrachten Sie die Funktion z = f(x,y), wobei x = x(t), y = y(t). Die Funktion z stellt eigentlich eine zusammengesetzte Funktion von *t* dar, wenn wir sie als z = f[x(t),y(t)] schreiben. In diesem Fall wird die Kettenregel für die Ableitung dz/dt wie folgt geschrieben:

$$\frac{\partial z}{\partial v} = \frac{\partial z}{\partial x} \cdot \frac{\partial x}{\partial v} + \frac{\partial z}{\partial y} \cdot \frac{\partial y}{\partial v}$$

Um den Ausdruck anzuzeigen, den der Taschenrechner für diese Version der Kettenregel erzeugt, geben Sie folgendes ein:



Das Ergebnis wird durch d1y(t)·d2z(x(t),y(t))+d1x(t)·d1z(x(y),y(t)) ausgegeben. Der Ausdruck d1y(t) bedeutet „die Ableitung von y(t) für die erste unabhängige Variable, d. h. t" oder d1y(t) = dy/dt. Entsprechend ist d1x(t) = dx/dt. Andererseits bedeutet d1z(x(t),y(t)) „die erste Ableitung von z(x,y) für die erste unabhängige Variable, d. h. x" oder d1z(x(t),y(t)) = $\partial z/\partial x$. Entsprechend ist d2z(x(t),y(t)) = $\partial z/\partial y$. Daher muss der obige Ausdruck folgendermaßen interpretiert werden:

$$dz/dt = (dy/dt)\cdot(\partial z/\partial y) + (dx/dt)\cdot(\partial z/\partial x).$$

Totales Differenzial einer Funktion z = z(x,y)

Wenn wir die letzte Gleichung mit dt multiplizieren, erhalten wir das totale Differenzial der Funktion z = z(x,y), d. h. $dz = (\partial z/\partial x)\cdot dx + (\partial z/\partial y)\cdot dy$.

Eine andere Variante der Kettenregel wird angewendet, wenn z = f(x,y), x = x(u,v), y = y(u,v), so dass z = f[x(u,v), y(u,v)]. Die folgenden Formeln stellen die Kettenregel für diesen Fall dar:

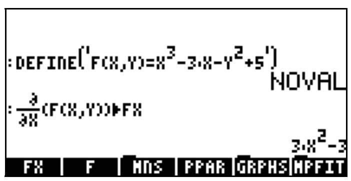
$$\frac{\partial z}{\partial u} = \frac{\partial z}{\partial x}\cdot\frac{\partial x}{\partial u} + \frac{\partial z}{\partial y}\cdot\frac{\partial y}{\partial u}, \qquad \frac{\partial z}{\partial v} = \frac{\partial z}{\partial x}\cdot\frac{\partial x}{\partial v} + \frac{\partial z}{\partial y}\cdot\frac{\partial y}{\partial v}$$

Bestimmen von Extremwerten von Funktionen mit zwei Variablen

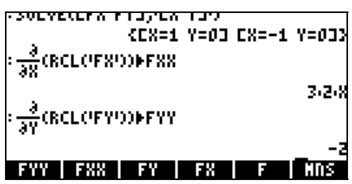
Damit die Funktion z = f(x,y) bei (x_o,y_o) einen Extrempunkt (Extremwert) aufweist, müssen ihre Ableitungen $\partial f/\partial x$ und $\partial f/\partial y$ an diesem Punkt verschwinden. Dies sind die notwendigen Bedingungen. Die hinreichenden Bedingungen dafür, dass die Funktion am Punkt (x_o,y_o) einen Extremwert aufweist, sind $\partial f/\partial x = 0$, $\partial f/\partial y = 0$ und $\Delta = (\partial^2 f/\partial x^2)\cdot(\partial^2 f/\partial y^2)-[\partial^2 f/\partial x\partial y]^2 > 0$. Der Punkt (x_o,y_o) ist ein relatives Maximum, wenn $\partial^2 f/\partial x^2 < 0$, oder ein relatives Minimum, wenn $\partial^2 f/\partial x^2 > 0$. Der Wert Δ wird als Diskriminante bezeichnet.

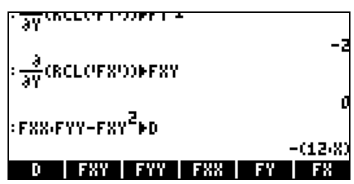
Wenn $\Delta = (\partial^2 f/\partial x^2)\cdot(\partial^2 f/\partial y^2)-[\partial^2 f/\partial x\partial y]^2 < 0$, liegt eine als *Sattelpunkt* bezeichnete Bedingung vor, wobei die Funktion bei x ein Maximum erreicht, wenn y als Konstante beibehalten wird, während gleichzeitig ein Minimum erreicht wird, wenn x als Konstante beibehalten wird, oder umgekehrt.

Beispiel 1 – Bestimmen Sie die Extrempunkte (sofern vorhanden) der Funktion $f(X,Y) = X^3-3X-Y^2+5$. Wir definieren zunächst die Funktion f(X/Y) und ihre Ableitungen $fX(X,Y) = \partial f/\partial X$, $fY(X,Y) = \partial f/\partial Y$. Anschließend lösen wir gleichzeitig die Gleichungen fX(X,Y) = 0 und fY(X,Y) = 0:

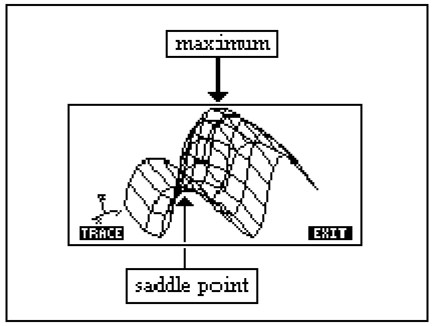


Wir finden kritische Punkte bei (X,Y) = (1,0) und (X,Y) = (-1,0). Zum Berechnen der Diskriminante berechnen wir die zweiten Ableitungen fXX(X,Y) = $\partial^2 f/\partial X^2$, fXY(X,Y) = $\partial^2 f/\partial X/\partial Y$ und fYY(X,Y) = $\partial^2 f/\partial Y^2$.





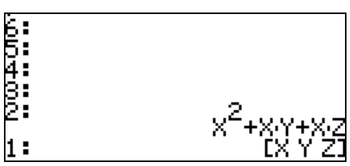
Das letzte Ergebnis gibt an, dass die Diskriminante $\Delta = -12X$ lautet. Für (X,Y) = (1,0) ist daher $\Delta<0$ (Sattelpunkt) und für (X,Y) = (-1,0) ist $\Delta>0$ und $\partial^2 f/\partial X^2<0$ (relatives Maximum). Die unten dargestellte, vom Taschenrechner erzeugte und mit dem Computer bearbeitete Abbildung veranschaulicht diese beiden Punkte:

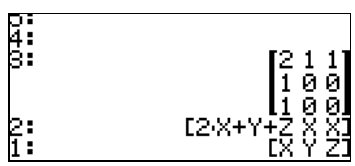


Verwenden der Funktion HESS zum Berechnen von Extremwerten

Die Funktion HESS kann wie folgt zum Berechnen der Extremwerte einer Funktion zweier Variablen verwendet werden. Als Eingabe für die Funktion HESS werden generell eine Funktion mit n unabhängigen Variablen $\phi(x_1, x_2, \ldots, x_n)$ und ein Vektor der Funktionen [„x_1“ „x_2“…„x_n“] verwendet. Die Funktion HESS gibt die Hesse-Matrix der Funktion ϕ zurück, definiert als die Matrix $\mathbf{H} = [h_{ij}] = [\partial^2\phi/\partial x_i \partial x_j]$, den Gradienten der Funktion für n Variablen **grad** f = [$\partial\phi/\partial x_1$, $\partial\phi/\partial x_2$, … $\partial\phi/\partial x_n$] und die Liste der Variablen [„x_1“ „x_2“…„x_n“].

Im RPN-Modus können Anwendungen der Funktion HESS einfacher grafisch dargestellt werden. Im folgenden Beispiel wenden wir die Funktion HESS auf die Funktion $\phi(X,Y,Z) = X^2 + XY + XZ$ an. Die Bildschirmabbildungen zeigen den RPN-Stack vor und nach der Anwendung der Funktion HESS.





Wenn der Gradient auf Ebene 2 auf eine Funktion mit zwei Variablen angewendet wird und gleich Null ist, stellt er die Gleichungen für kritische Punkte dar, d. h. $\partial\phi/\partial x_i = 0$, während die Matrix auf Ebene 3 Ableitungen zweiter Ordnung darstellt. Somit können die Ergebnisse der Funktion HESS zum Berechnen der Extremwerte von Funktionen mit zwei Variablen verwendet werden. Gehen Sie beispielsweise für die Funktion $f(X,Y) = X^3-3X-Y^2+5$ im RPN-Modus wie folgt vor:

X^3-3*X-Y^2+5' `ENTER` ['X','Y'] `ENTER`	Funktion und Variablen eingeben
HESS	Funktion HESS anwenden
SOLVE	Kritische Punkte suchen
`EVAL`	Vektor zerlegen
's1' `STO▸` 's2' `STO▸`	Kritische Punkte speichern

Die Variablen s1 und s2 enthalten an dieser Stelle die Vektoren [„X=-1",„Y=0"] bzw. [„X=1",„Y=0"]. Die Hesse-Matrix befindet sich an dieser Stelle auf Ebene 1.

'H' `STO▸`	Hesse-Matrix speichern
`VAR` `H` `S1` SUBST `→` `→NUM`	s1 in H ersetzen

Die resultierende Matrix **A** besitzt a_{11} Elemente $a_{11} = \partial^2\phi/\partial X^2 = -6.$, $a_{22} = \partial^2\phi/\partial X^2 = -2.$ und $a_{12} = a_{21} = \partial^2\phi/\partial X\partial Y = 0$. Die Diskriminante für diesen kritischen Punkt s1(-1,0) ist $\Delta = (\partial^2 f/\partial x^2)\cdot(\partial^2 f/\partial y^2)-[\partial^2 f/\partial x\partial y]^2 = (-6.)(-2.) = 12.0 > 0$. Da $\partial^2\phi/\partial X^2 < 0$, stellt Punkt s1 ein relatives Maximum dar.

Anschließend ersetzen wir den zweiten Punkt s2 in H:

`VAR` `H` `S2` SUBST `→` `→NUM`	s2 in H ersetzen

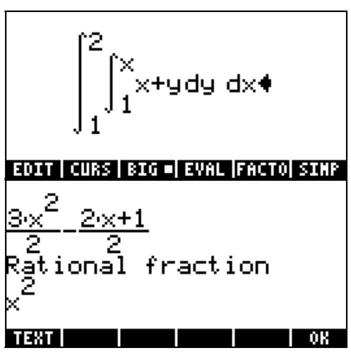
Die resultierende Matrix besitzt a_{11} Elemente $a_{11} = \partial^2\phi/\partial X^2 = 6.$, $a_{22} = \partial^2\phi/\partial X^2 = -2.$ und $a_{12} = a_{21} = \partial^2\phi/\partial X\partial Y = 0$. Die Diskriminante für diesen kritischen Punkt s2(1,0) ist $\Delta = (\partial^2 f/\partial x^2)\cdot(\partial^2 f/\partial y^2)-[\partial^2 f/\partial x\partial y]^2 = (6.)(-2.) -12.0 < 0$ und gibt einen Sattelpunkt an.

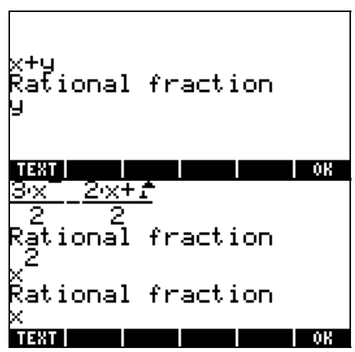
Mehrfache Integrale

Eine physische Interpretation eines normalen Integrals $\int_a^b f(x)dx$ ist die Fläche unter der Kurve y = f(x) und den x-Koordinaten x = a und x = b. Die Erweiterung eines normalen Integrals auf drei Dimensionen ist ein doppeltes Integral einer Funktion f(x,y) über einem Bereich R auf der x-y-Fläche, das den Rauminhalt des Körpers unter der Fläche f(x,y) über dem Bereich R darstellt. Der Bereich R kann als R = {a<x<b, f(x)<y<g(x)} oder als R = {c<y<d, r(y)<x<s(y)} beschrieben werden. Somit kann das doppelte Integral wie folgt geschrieben werden:

$$\iint_R \phi(x,y)dA = \int_a^b \int_{f(x)}^{g(x)} \phi(x,y)dydx = \int_c^d \int_{r(y)}^{s(y)} \phi(x,y)dydx$$

Die Berechnung eines doppelten Integrals mit dem Taschenrechner ist unkompliziert. Ein doppeltes Integral kann in EquationWriter erzeugt werden (siehe das Beispiel in Kapitel 2). Ein Beispiel folgt. Dieses doppelte Integral wird direkt in EquationWriter berechnet, indem der ganze Ausdruck ausgewählt und die Funktion EVAL verwendet wird. Das Ergebnis ist 3/2. Wenn Sie im CAS MODES-Fenster die Option Step/Step auswählen, ist eine schrittweise Ausgabe möglich.





Jacobimatrix einer Koordinatentransformation

Betrachten Sie die Koordinatentransformation x = x(u,v), y = y(u,v). Die Jacobimatrix dieser Transformation wird definiert als

$$|J| = \det(J) = \det\begin{pmatrix} \frac{\partial x}{\partial u} & \frac{\partial x}{\partial v} \\ \frac{\partial y}{\partial u} & \frac{\partial y}{\partial v} \end{pmatrix}$$

Wenn Sie mit dieser Transformation ein Integral berechnen, lautet der zu verwendende Ausdruck $\iint_R \phi(x,y)dydx = \iint_{R'} \phi[x(u,v), y(u,v)]\,|J|\,dudv$,

wobei R' der durch die Koordinaten (u,v) angegebene Bereich R ist.

Doppeltes Integral in Polarkoordinaten

Zur Transformation von Polarkoordinaten zu kartesischen Koordinaten verwenden wir x(r,θ) = r cos θ und y(r, θ) = r sin θ. Somit lautet die Jacobimatrix der Transformation

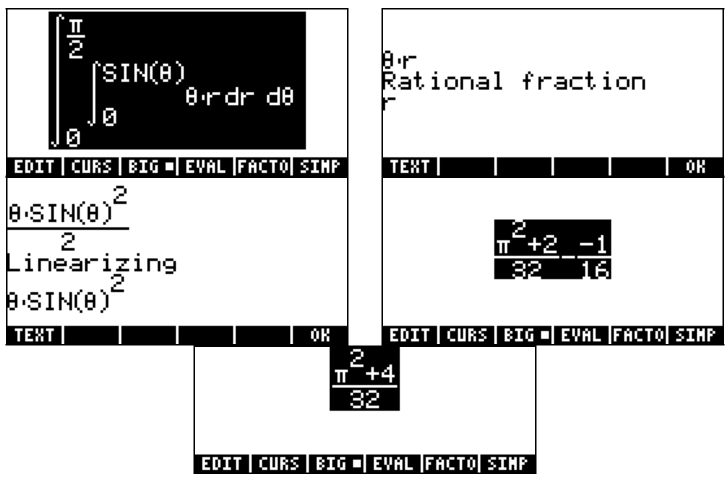
$$|J| = \begin{vmatrix} \frac{\partial x}{\partial r} & \frac{\partial x}{\partial \theta} \\ \frac{\partial y}{\partial r} & \frac{\partial y}{\partial \theta} \end{vmatrix} = \begin{vmatrix} \cos(\theta) & -r\cdot\sin(\theta) \\ \sin(\theta) & r\cdot\cos(\theta) \end{vmatrix} = r$$

Bei diesem Ergebnis werden Integrale in Polarkoordinaten als

$$\iint_{R'} \phi(r,\theta)dA = \int_{\alpha}^{\beta}\int_{f(\theta)}^{g(\theta)} \phi(r,\theta)rdrd\theta$$

geschrieben, wobei der Bereich R' in Polarkoordinaten R' = {$\alpha < \theta < \beta$, $f(\theta) < r < g(\theta)$} lautet.

Doppelte Integrale in Polarkoordinaten können in den Taschenrechner eingegeben werden, wenn sichergestellt wird, dass die Jacobimatrix |J| = r im Integranden enthalten ist. Es folgt ein Beispiel für ein doppeltes Integral, dessen Berechnung in Polarkoordinaten schrittweise angezeigt wird:



Kapitel 15
Anwendungen der Vektorrechnung

In diesem Kapitel stellen wir mehrere Funktionen im Menü CALC für die Berechnung von Skalar- und Vektorfeldern vor. Das Menü CALC wurde in Kapitel 13 ausführlich dargestellt. Wir wiesen insbesondere auf mehrere Funktionen im Menü DERIV&INTEG hin, die für die Vektorrechnung verwendet werden können, nämlich CURL, DIV, HESS und LAPL. Ändern Sie für die Übungen in diesem Kapitel das Winkelmaß in Radiant.

Definitionen

Eine für einen Raumbereich definierte Funktion, z. B. $\phi(x,y,z)$, wird als Skalarfeld bezeichnet. Beispiele hierfür sind Temperatur, Dichte und Spannung in der Nähe einer Ladung. Wenn die Funktion durch einen Vektor definiert ist, d. h. $\mathbf{F}(x,y,z) = f(x,y,z)\mathbf{i}+g(x,y,z)\mathbf{j}+h(x,y,z)\mathbf{k}$, wird sie als Vektorfeld bezeichnet.

Der folgende Operator, der als Del- oder Nabla-Operator bezeichnet wird, ist ein Vektor-Operator, der auf eine Skalar- oder Vektorfunktion angewendet werden kann:

$$\nabla[\ ] = i \cdot \frac{\partial}{\partial x}[\ ] + j \cdot \frac{\partial}{\partial y}[\ ] + k \cdot \frac{\partial}{\partial z}[\ ]$$

Wenn dieser Operator auf eine Skalarfunktion angewendet wird, können wir den Gradienten der Funktion erhalten, und wenn er auf eine Vektorfunktion angewendet wird, können wir die Divergenz und die Rotation dieser Funktion erhalten. Die Kombination von Gradient und Divergenz ergibt einen weiteren Operator, der als Laplace-Operator einer Skalarfunktion bezeichnet wird. Diese Operationen werden nun dargestellt.

Gradient und Richtungsableitung

Der <u>Gradient</u> einer Skalarfunktion $\phi(x,y,z)$ ist eine Vektorfunktion, die durch

$$grad\phi = \nabla\phi = i \cdot \frac{\partial\phi}{\partial x} + j \cdot \frac{\partial\phi}{\partial y} + k \cdot \frac{\partial\phi}{\partial z}$$

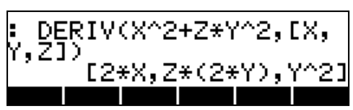
definiert ist. Das skalare Produkt des Gradienten einer Funktion mit einem bestimmten Einheitsvektor stellt die Änderungsrate der Funktion entlang diesem bestimmten Vektor dar.Diese Änderungsrate wird als Richtungsableitung $D_u\phi(x,y,z) = \mathbf{u}\bullet\nabla\phi$ der Funktion bezeichnet.

Die maximale Änderungsrate der Funktion erfolgt an jedem beliebigen Punkt in der Richtung des Gradienten, d. h. entlang einem Einheitsvektor $\mathbf{u} = \nabla\phi/|\nabla\phi|$.

Der Wert der Richtungsableitung ist gleich dem Betrag des Gradienten an einem beliebigen Punkt $D_{max}\phi(x,y,z) = \nabla\phi \bullet\nabla\phi/|\nabla\phi| = |\nabla\phi|$.

Die Gleichung $\phi(x,y,z) = 0$ stellt eine Fläche im Raum dar. Der Gradient der Funktion ist an jedem Punkt der Fläche senkrecht zur Fläche. Daher kann die Gleichung der Tangentialebene zur Kurve an diesem Punkt mit der in Kapitel 9 dargestellten Methode ermittelt werden.

Die im Menü CALC verfügbare Funktion DERIV stellt die einfachste Möglichkeit dar, den Gradienten zu erhalten, z. B.

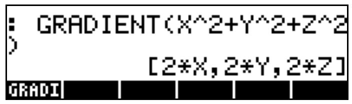


Ein Programm zum Berechnen des Gradienten

Das folgende Programm, das Sie in der Variablen GRADIENT speichern können, verwendet die Funktion DERIV zum Berechnen des Gradienten einer Skalarfunktion von X,Y,Z. Berechnungen für andere Basisvariablen sind nicht möglich. Wenn Sie häufig mit (X,Y,Z) arbeiten, erleichtert diese Funktion jedoch die Berechnungen:

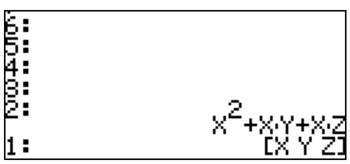
<< X Y Z 3 →ARRY DERIV >>

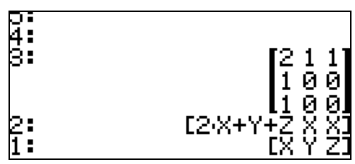
Geben Sie das Programm im RPN-Modus ein. Nachdem Sie den ALG-Modus gestartet haben, können Sie die Funktion GRADIENT wie im folgenden Beispiel aufrufen:



Verwenden der Funktion HESS zum Erhalten des Gradienten

Mit der Funktion HESS können Sie den Gradienten einer Funktion wie im Folgenden dargestellt erhalten. Wie in Kapitel 14 erläutert, wird als Eingabe für die Funktion HESS eine Funktion mit n unabhängigen Variablen $\phi(x_1, x_2, \ldots, x_n)$ und ein Vektor der Funktionen [„x_1“ „x_2“…„x_n“] verwendet. Die Funktion HESS gibt die Hesse-Matrix der Funktion ϕ zurück, definiert als die Matrix $\mathbf{H} = [h_{ij}] = [\partial^2\phi/\partial x_i \partial x_j]$, den Gradienten der Funktion für n Variablen **grad** f = [$\partial\phi/\partial x_1$, $\partial\phi/\partial x_2$, … $\partial\phi/\partial x_n$] und die Liste der Variablen [„x_1“ „x_2“…„x_n“]. Im folgenden Beispiel wenden wir im RPN-Modus die Funktion HESS auf das Skalarfeld $\phi(X,Y,Z) = X^2 + XY + XZ$ an:





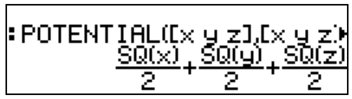
Somit ist der Gradient [2X+Y+Z, X, X].

Potential eines Gradienten

Wenn ein Vektorfeld $\mathbf{F}(x,y,z) = f(x,y,z)\mathbf{i}+g(x,y,z)\mathbf{j}+h(x,y,z)\mathbf{k}$ und eine Funktion $\phi(x,y,z)$ gegeben sind, so dass $f = \partial\phi/\partial x$, $g = \partial\phi/\partial y$ und $h = \partial\phi/\partial z$, dann wird $\phi(x,y,z)$ als Potentialfunktion für das Vektorfeld **F** bezeichnet. Es folgt, dass $\mathbf{F} = \text{grad}\ \phi = \nabla\phi$.

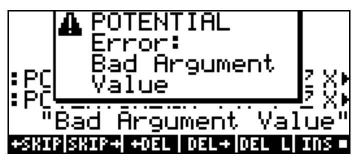
Der Taschenrechner enthält die Funktion POTENTIAL, die über den Befehlskatalog (↱ _CAT_) verfügbar ist, um die Potentialfunktion eines Vektorfeldes zu berechnen, sofern vorhanden. Wenn beispielsweise $\mathbf{F}(x,y,z) =$

xi + y**j** + z**k** ist, ergibt sich durch Anwenden der Funktion POTENTIAL Folgendes:



Da die Funktion SQ(x) den Wert x^2 darstellt, gibt dieses Ergebnis an, dass die Potentialfunktion für das Vektorfeld **F**(x,y,z) = x**i** + y**j** + z**k** durch die Gleichung $\phi(x,y,z) = (x^2+y^2+z^2)/2$ dargestellt wird.

Beachten Sie, dass die Bedingungen für das Vorhandensein von $\phi(x,y,z)$, nämlich $f = \partial\phi/\partial x$, $g = \partial\phi/\partial y$ und $h = \partial\phi/\partial z$, mit den Bedingungen $\partial f/\partial y = \partial g/\partial x$, $\partial f/\partial z = \partial h/\partial x$ und $\partial g/\partial z = \partial h/\partial y$ äquivalent sind. Anhand dieser Bedingungen lässt sich schnell bestimmen, ob das Vektorfeld über eine entsprechende Potentialfunktion verfügt. Wenn eine der Bedingungen $\partial f/\partial y = \partial g/\partial x$, $\partial f/\partial z = \partial h/\partial x$, $\partial g/\partial z = \partial h/\partial y$ nicht zutrifft, ist die Potentialfunktion $\phi(x,y,z)$ nicht vorhanden. In diesem Fall gibt die Funktion POTENTIAL eine Fehlermeldung zurück. Beispielsweise weist das Vektorfeld **F**(x,y,z) = (x+y)**i** + (x-y+z)**j** + xz**k** keine entsprechende Potentialfunktion auf, da $\partial f/\partial z \neq \partial h/\partial x$. Die Meldung des Taschenrechners in diesem Fall wird unten dargestellt:

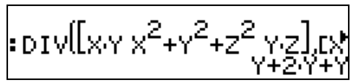


Divergenz

Die Divergenz einer Vektorfunktion **F**(x,y,z) = f(x,y,z)**i**+g(x,y,z)**j**+h(x,y,z)**k** wird als "skalares Produkt" des Del-Operators mit der Funktion bezeichnet, d. h.

$$divF = \nabla \bullet F = \frac{\partial f}{\partial x} + \frac{\partial g}{\partial y} + \frac{\partial h}{\partial z}$$

Die Divergenz eines Vektorfeldes kann mit der Funktion DIV berechnet werden. Beispielsweise wird für **F**(X,Y,Z) = [XY,X^2+Y^2+Z^2,YZ] die Divergenz wie folgt im ALG-Modus berechnet:

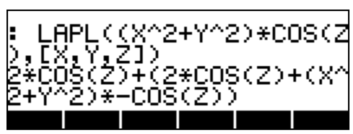


Laplace-Operator

Die Divergenz des Gradienten einer Skalarfunktion ergibt einen Operator, der als Laplace-Oper bezeichnet wird. Der Laplace-Operator einer Skalarfunktion ϕ(x,y,z) wird somit durch

$$\nabla^2\phi = \nabla \bullet \nabla\phi = \frac{\partial^2\phi}{\partial x^2} + \frac{\partial^2\phi}{\partial x^2} + \frac{\partial^2\phi}{\partial x^2}$$

angegeben. Die partielle Differenzialgleichung $\nabla^2\phi$ = 0 wird als Laplace-Gleichung bezeichnet.
Der Laplace-Operator einer Skalarfunktion kann mit der Funktion LAPL berechnet werden. Geben Sie beispielsweise zum Berechnen des Laplace-Operators der Funktion ϕ(X,Y,Z) = (X^2+Y^2)cos(Z) Folgendes ein:



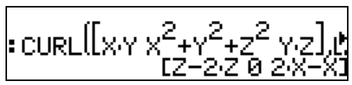
Rotation

Die Rotation eines Vektorfeldes **F**(x,y,z) = f(x,y,z)**i**+g(x,y,z)**j**+h(x,y,z)**k** wird durch das Kreuzprodukt des Del-Operators mit dem Vektorfeld definiert, d. h.

$$curl\mathbf{F} = \nabla \times \mathbf{F} = \begin{vmatrix} \mathbf{i} & \mathbf{j} & \mathbf{k} \\ \frac{\partial}{\partial x}[\ ] & \frac{\partial}{\partial y}[\ ] & \frac{\partial}{\partial z}[\ ] \\ f(x,y,z) & g(x,y,z) & h(x,y,z) \end{vmatrix}$$

$$= \mathbf{i}\left(\frac{\partial h}{\partial y} - \frac{\partial g}{\partial z}\right) + \mathbf{j}\left(\frac{\partial f}{\partial z} - \frac{\partial h}{\partial x}\right) + \mathbf{k}\left(\frac{\partial h}{\partial y} - \frac{\partial g}{\partial z}\right)$$

Die Rotation des Vektorfeldes kann mit der Funktion CURL berechnet werden. Beispielsweise wird für die Funktion **F**(X,Y,Z) = [XY,X^2+Y^2+Z^2,YZ] die Rotation wie folgt berechnet:



Rotationsfreie Felder und Potentialfunktion

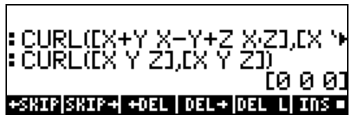
In einem vorherigen Abschnitt dieses Kapitels stellten wir die Funktion POTENTIAL vor, um die Potentialfunktion φ(x,y,z) für eine Vektorfeld **F**(x,y,z) = f(x,y,z)**i**+ g(x,y,z)**j**+ h(x,y,z)**k** zu berechnen, so dass **F** = grad φ = ∇φ.Wir gaben auch an, dass die Bedingungen für das Vorhandensein von φ wie folgt lauten: ∂f/∂y = ∂g/∂x, ∂f/∂z = ∂h/∂x und ∂g/∂z = ∂h/∂y. Diese Bedingungen sind mit folgendem Vektorausdruck äquivalent:

$$\text{curl } \mathbf{F} = \nabla\times\mathbf{F} = 0.$$

Ein Vektorfeld **F**(x,y,z) mit der Rotation Null wird als rotationsfreies Feld bezeichnet. Daher schließen wir, dass für eine Potentialfunktion φ(x,y,z) stets ein rotationsfreies Feld **F**(x,y,z) vorhanden ist.

In einem vorherigen Beispiel versuchten wir, eine Potentialfunktion für das Vektorfeld **F**(x,y,z) = (x+y)**i** + (x-y+z)**j** + xz**k** zu finden und erhielten eine von der Funktion POTENTIAL zurückgegebene Fehlermeldung. Um zu überprüfen, ob es sich hierbei um ein Rotationsfeld handelt (d. h. ∇×**F** ≠ 0), verwenden wir für dieses Feld die Funktion CURL:

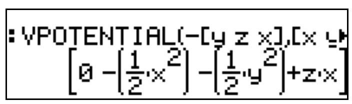
Andererseits ist das Vektorfeld **F**(x,y,z) = **xi** + y**j** + z**k** tatsächlich rotationsfrei, wie unten gezeigt:



Vektorpotential

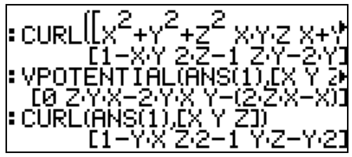
Wenn für ein Vektorfeld **F**(x,y,z) = f(x,y,z)**i**+g(x,y,z)**j**+h(x,y,z)**k** eine Vektorfunktion Φ(x,y,z) = ϕ(x,y,z)**i**+ψ(x,y,z)**j**+η(x,y,z)**k** vorhanden ist, so dass **F** = curl Φ = $\nabla\times\Phi$, wird die Funktion Φ(x,y,z) als Vektorpotential von **F**(x,y,z) bezeichnet.

Der Taschenrechner enthält die über den Befehlskatalog (↱ CAT) verfügbare Funktion VPOTENTIAL, um das Vektorpotential Φ(x,y,z) zu berechnen, wenn das Vektorfeld **F**(x,y,z) = f(x,y,z)**i**+g(x,y,z)**j**+h(x,y,z)**k** vorhanden ist. Für das Vektorfeld **F**(x,y,z) = -(y**i**+z**j**+x**k**) ergibt VPOTENTIAL beispielsweise



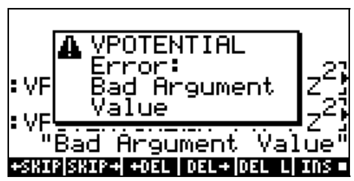
d. h. Φ(x,y,z) = $-x^2/2$**j** + ($-y^2/2+zx$)**k**.

Es sollte darauf hingewiesen werden, dass für ein Vektorfeld **F** mehrere Vektorpotentialfunktionen Φ vorhanden sind. In der folgenden Bildschirmabbildung wird beispielsweise gezeigt, dass die Rotation der Vektorfunktion Φ_1 = [$X^2+Y^2+Z^2$,XYZ,X+Y+Z] durch den Vektor **F** = $\nabla\times\Phi_2$ = [1-XY,2Z-1,ZY-2Y] dargestellt wird. Durch Anwendung der Funktion VPOTENTIAL wird die Vektorpotentialfunktion Φ_2 = [0, ZYX-2YX, Y-(2ZX-X)] erstellt, die sich von Φ_1 unterscheidet. Der letzte Befehl in der Bildschirmabbildung zeigt, dass tatsächlich **F** = $\nabla\times\Phi_2$. Somit ist eine Vektorpotentialfunktion nicht eindeutig bestimmt.

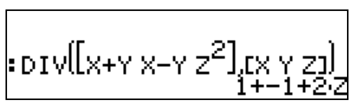


Die Komponenten des Vektorfeldes **F**(x,y,z) = f(x,y,z)**i**+g(x,y,z)**j** +h(x,y,z)**k** und der Vektorpotentialfunktion Φ(x,y,z) = φ(x,y,z)**i**+ψ(x,y,z)**j**+η(x,y,z)**k** stehen durch f = ∂η/∂y - ∂ψ/∂x, g = ∂φ/∂z - ∂η/∂x und h = ∂ψ/∂x - ∂φ/∂y miteinander in Beziehung.

Als Bedingung für das Vorhandensein der Funktion Φ(x,y,z) ist **F** = ∇•**F** = 0, d. h. ∂f/∂x + ∂g/∂y + ∂f/∂z = 0. Wenn daher diese Bedingung nicht erfüllt ist, ist die Vektorpotentialfunktion Φ(x,y,z) nicht vorhanden. Beispielsweise gibt die Funktion VPOTENTIAL für **F** = [X+Y,X-Y,Z^2] eine Fehlermeldung zurück, da die Funktion F die Bedingung ∇•**F** = 0 nicht erfüllt:



Die Bedingung ∇•**F** ≠ 0 wird in der folgenden Bildschirmabbildung überprüft:



Kapitel 16
Differentialgleichungen

In diesem Kapitel stellen wir Beispiele dar des Lösens der gewöhnlichen Differentialgleichungen (ODE) Rechnerfunktionen verwendend. Eine Differentialgleichung ist eine Gleichung, die Ableitungen der unabhängigen Variable mit einbezieht. In den meisten Fällen suchen wir die abhängige Funktion, die die Differentialgleichung erfüllt.

Grundlegende Betriebe mit Differentialgleichungen

In diesem Abschnitt stellen wir etwas Gebrauch des Rechners für das Eintragen, die Überprüfung und das Sichtbar machen der Lösung von ODEs dar.

Hereinkommende Differentialgleichungen

Der Schlüssel zum Verwenden von von Differentialgleichungen im Rechner schreibt in den Ableitungen in der Gleichung. Die einfachste Weise, eine Differentialgleichung einzutragen soll sie im Gleichung Verfasser schreiben. Z.B. das folgende ODE schreiben:

$(x-1)\times(dy/dx)^2 + 2xy = e^{-x}$ Sin x, Gebrauch:

[↱] EQW [↰] ( ) [ALPHA] [↰] [X] [−] [1] [▶] [▶] [▶] [×] [↱] ∂ [ALPHA] [↰] [X]
[▶] [ALPHA] [↰] [Y] [↰] ( ) [ALPHA] [↰] [X] [▶] [▶] [Y^x] [2] [▶] [▶] [+] [2] [×]
[ALPHA] [↰] [X] [×] [ALPHA] [↰] [Y] [↰] ( ) [ALPHA] [↰] [X] [▶] [▶] [▶] [▶]
[↱] = [↰] e^x [ALPHA] [↰] [X] [▶] [×] [SIN] [ALPHA] [↰] [X] [ENTER]

Die Ableitung dy/dx wird durch ∂ x(y(x) dargestellt). Für Lösung oder Berechnung Zwecke müssen Sie y(x) im Ausdruck spezifizieren, d.h. muß die abhängige Variable sein unabhängiges variable(s) in jeder möglicher Ableitung in der Gleichung umfassen.
Sie können eine Gleichung direkt in einstapeln auch schreiben, indem Sie das Symbol in den Ableitungen verwenden. Z.B. das folgende ODE schreiben, das zweitrangige Ableitungen mit einbezieht: $d^2u/dx^2 + 3 \cdot u \cdot (du/dx) + u2 = 1/x$, direkt in einstapeln, Gebrauch:

['] [↱] ∂ [ALPHA] [↰] [X] [↰] () [↱] ∂ [ALPHA] [↰] [X] [↰] () [ALPHA] [↰] [U]
[↰] () [ALPHA] [↰] [X] [▶] [▶] [▶] [+] [3] [×] [ALPHA] [↰] [U] [↰] ()
[ALPHA] [↰] [X] [▶] [×] [↱] ∂ [ALPHA] [↰] [X] [↰] () [ALPHA] [↰] [U] [↰] ()
[ALPHA] [↰] [X] [▶] [▶] [+] [ALPHA] [↰] [U] [↰] () [ALPHA] [↰] [X] [▶] [Y^x] [2]
[↱] = [1] [÷] [ALPHA] [↰] [X] [ENTER]

Das Resultat ist '∂x(∂x(u(x)))+3*u(x)*∂x(u(x))+u^2=1/x '. Dieses Format zeigt oben im Schirm wenn die _ Textbolk Wahl in der Anzeige Einstellung ([MODE] DISP) wird nicht vorgewählt. Betätigen Sie [▼], um die Gleichung im Gleichung Verfasser zu sehen.

Eine alternative Darstellung für Ableitungen schrieb direkt im Stapel soll ' d1 ' für die Ableitung in Bezug auf die erste unabhängige Variable, ' d2 ' für die Ableitung in Bezug auf die zweite unabhängige Variable verwenden, das usw.. Eine zweitrangige Ableitung z.B. d^2x/dt^2, in der x = x(t), als ' d1d1x(t) ', während geschrieben würde (dx/dt)2 würde ' d1x(t)^2 ' geschrieben. So das PDE 2y/t2 - g(x, y)× (2y/x2)2 = r(x, y), würden mit dieser Darstellung, wie ' d2d2y(x, t)-g(x, y)*d1d1y(x, t)^2=r(x, y) ' geschrieben.

Die Darstellung mit ' d ' und dem Auftrag der unabhängigen Variable ist die Darstellung, die durch den Rechner bevorzugt wird, wenn Ableitungen in eine Berechnung miteinbezogen werden. Z.B. produziert das Verwenden der Funktion DERIV, im ALG Modus, wie gezeigtem folgendem DERIV('x*f(x, t)+g(t, y) = h(x, y, t)', t), den folgenden Ausdruck:
' x*d2f(x, t)+d1g(t, y)=d3h(x, y, t) '. Übersetzt zum Papier, stellt dieser Ausdruck die teilweise Differentialgleichun $x\cdot(\partial f/\partial t) + \partial g/\partial t = \partial h/\partial t$.

Weil der Auftrag des variablen t im f(x, in t), im g(t, in y) und in h(x, y, t) unterschiedlich ist, haben Ableitungen in Bezug auf t unterschiedliche Indizes d.h. d2f(x, t), d1g(t, y) und d3h(x, y, t). Alle stellen jedoch Ableitungen in Bezug auf die gleiche Variable dar.

Ausdrücke für die Ableitungen, welche die Auftrag-von-variable Indexdarstellung verwenden, übersetzen nicht in abgeleitete Darstellung im Gleichung Verfasser, wie Sie überprüfen können, indem Sie [▼] betätigen, während das letzte Resultat in Stapelniveau 1 ist. Jedoch versteht der Rechner beide Darstellungen und läßt der benutzten Darstellung dementsprechend betrachten laufen.

Überprüfung der Lösungen im Rechner

, ob eine Funktion eine bestimmte Gleichung mit dem Rechner, Gebrauchfunktion SUBST (siehe Kapitel 5), die Lösung in der Form zu ersetzen ' y = f(x) ' oder ' y = f(x, t) ', etc., erfüllen in die Differentialgleichung überprüfen. Sie können das Resultat vereinfachen müssen, indem Sie Funktion EVAL verwenden, um die Lösung zu überprüfen. Z.B. um zu prüfen, ob u = A Sünde wot eine Lösung der Gleichung d2u/dt2 + wo2×u = 0 ist, verwenden Sie das folgende:
Im ALG Modus:

SUBST('∂t(∂t(u(t)))+ ω0^2*u(t) = 0','u(t) =A*SIN (ω0*t) ' `ENTER
EVAL(ANS(1)) ENTER

Im RPN Modus:

'∂t(∂t(u(t)))+ ω0^2*u(t) = 0' ENTER ' u(t)=A*SIN (ω0*t) ' ENTER
SUBST EVAL

Das Resultat ist ' 0=0 '.

Für dieses Beispiel konnten Sie auch verwenden: '∂t(∂t(u(t))))+ ω0^2*u(t) = 0', zum der Differentialgleichung einzutragen.

Steigungfeldsichtbarmachung der Lösungen

Die Steigungparzellen, eingeführt in Kapitel 12, werden benutzt, um die Lösungen zu einer Differentialgleichung der Form sichtbar zu machen dy/dx = f(x, y). Eine Steigungparzelle zeigt eine Anzahl von den Segmenten, die zu den Lösung Kurven, y = f(x) tangential sind. Die Steigung der Segmente an irgendeinem Punkt (x,y) wird durch dy/dx = f(x, y) gegeben, ausgewertet an irgendeinem Punkt (x,y), darstellt die Steigung der Tangentelinie am Punkt (x,y).

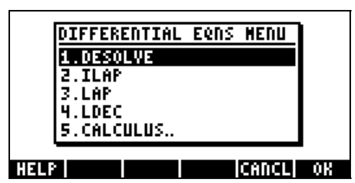
Beispiel 1 -- Vollziehen Sie die Lösung zur Differentialgleichung y ' = f(x, y) = Sin x Lattich y mit einer Steigungparzelle nach. Um dieses Problem zu lösen, befolgen Sie die Anweisungen in Kapitel 12 für slopefield Plots.

Wenn Sie die Steigungparzelle im Papier reproduzieren konnten, können Sie Linien eigenhändig verfolgen, die Tangente zu den Strichen sind, die im Plot gezeigt werden. Dieses zeichnet festsetzen Linien von y(x, y) = Konstante, für die Lösung von y ' = f(x, y). So sind Steigungfelder nützliche Werkzeuge für das Sichtbar machen besonders der schwierigen Gleichungen, um zu lösen.

In der Zusammenfassung sind Steigungfelder die graphischen Hilfsmittel, zum der Kurven zu skizzieren y = g(x), die Lösungen der Differentialgleichung dy/dx = f(x,y).

Das CALC/DIFF Menü

Das Unterprogramm des DIFFERENTIALS EQNS.. innerhalb des CALC (CALC) Menüs stellt Funktionen für die Lösung von Differentialgleichungen zur Verfügung. Das Menü wird nachstehend mit Systemmarkierungsfahne 117 Satz aufgeführt, um Kästen CHOOSE:



Diese Funktionen werden kurz zunächst beschrieben. Sie werden ausführlicher in den neueren Teilen dieses Kapitels beschrieben.

DESOLVE: Differentialgleichung-Löser, stellt eine Lösung zur Verfügung, wenn möglich
ILAP : Umgekehrtes LAPlace wandeln, L-1[F(s)] = f(t) um
LAP : LAPlace wandeln, L[f(t)]=F(s) um
LDEC : löst lineare Differentialgleichungen mit konstanten Koeffizienten, einschließlich Systeme von Differentialgleichungen mit konstanten Koeffizienten.

Lösung zu den linearen und nicht linearen Gleichungen

Eine Gleichung, in der die abhängige Variable und alle seine passenden Ableitungen vom ersten Grad sind, gekennzeichnet als eine <u>lineare Differentialgleichung</u>. Andernfalls soll die Gleichung <u>nicht linear</u>. Beispiele der linearen Differentialgleichungen sind: $d^2x/dt^2 + \beta\cdot(dx/dt) + \omega_o\cdot x = A \sin \omega_f t$, and $\partial C/\partial t + u\cdot(\partial C/\partial x) = D\cdot(\partial^2 C/\partial x^2)$.

Eine Gleichung deren rechte Seite (die Funktion oder seine Ableitungen nicht, mit einbeziehend) bis null gleich ist, wird eine homogene Gleichung genannt. Andernfalls wird sie nicht homogen genannt. Die Lösung zur homogenen Gleichung bekannt als allgemeine Lösung. Eine bestimmte Lösung ist eine, die die nicht homogene Gleichung erfüllt.

Funktion LDEC

Der Rechner liefert Funktion LDEC (linearer Differentialgleichung-Befehl) um die allgemeine Lösung zu einer linearen ODE jedes möglichen Auftrages mit konstanten Koeffizienten zu finden, ob er oder nicht homogen ist. Diese Funktion erfordert Sie, zwei Stücke Eingang zur Verfügung zu stellen:

· die rechte Seite der ODE
· die charakteristische Gleichung der ODE

Beide dieser Eingänge müssen in der Rückstellung unabhängigen Variable für CAS gegeben werden des Rechners (gewöhnlich X) ausgedrückt. Der Ausgang von der Funktion ist die allgemeine Lösung der ODE. Die Funktion LDEC ist durch im CALC/DIFF Menü vorhanden. Die Beispiele werden im RPN Modus gezeigt, sie zum ALG Modus jedoch ist zu übersetzen direkt.

<u>Beispiel 1</u> - Die homogene ODE lösen: d3y/dx3-4×(d2y/dx2)-11×(dy/dx)+30×y = 0, kommen herein: 0 ` ' X^3-4*X^2-11*X+30 ' ` LDEC. Die Lösung ist:

$$-\frac{6\cdot cC0-(cC1+cC2)}{24}\cdot e^{5\cdot X}+\frac{10\cdot cC0-(7\cdot cC1-cC2)}{40}\cdot e^{-(3\cdot X)}+\frac{15\cdot cC0+2\cdot cC1-cC2}{15}\cdot e^{2\cdot X}$$

wo cC0, cC1 und cC2 Integrationskonstanten sind. Während dieses Resultat sehr schwierig scheint, kann es vereinfacht werden, wenn wir nehmen

K1 = (10*C0-(7+C1-C2))/40, K2 = -(6*C0-(C1+C2))/24,

und

K3 = (15*C0+(2*C1-C2))/15.

Dann ist die Lösung

$$y = K_1 \cdot e^{-3x} + K_2 \cdot e^{5x} + K_3 \cdot e^{2x}.$$

Der Grund, warum das Resultat, das von LDEC zur Verfügung gestellt wird, solche schwierige Kombination von Konstanten zeigt, ist, weil innerlich die Lösung zu produzieren, LDEC Laplace verwendet, umwandelt (in diesem Kapitel später dargestellt werden), die die Lösung eines ODE in eine algebraische Lösung umwandeln. Die Kombination von Konstanten resultieren aus dem Faktor aus den exponentialen Bezeichnungen, nachdem das Laplace Lösung ist erreicht umwandeln.

Beispiel 2 - Mit der Funktion LDEC, lösen Sie das nicht homogene ODE:

$$d^3y/dx^3-4\cdot(d^2y/dx^2)-11\cdot(dy/dx)+30\cdot y = x^2.$$

Kommen Sie herein:

'X^2' ENTER 'X^3-4*X^2-11*X+30' ENTER LDEC

Die Lösung ist:

$$-\frac{750\cdot cC0-(125\cdot cC1+125\cdot cC2+2)}{3000}\cdot e^{5\cdot X}+\frac{270\cdot cC0-(189\cdot cC1-(27\cdot cC2-2))}{1080}\cdot e\cdot e^{-(3\cdot X)}+\frac{450\cdot X^2+330\cdot X+241}{13500}\cdot e^{(}$$

Die Kombination der Konstanten, welche die exponentialen Bezeichnungen mit einfacheren Werten, wie ersetzend, ergibt K_3 begleiten, = -(750*C0-(125*C1+125*C2+2))/3000, den Ausdruck $y = K_1 \cdot e^{-3x} + K_2 \cdot e^{5x} + K_3 \cdot e^{2x} + (450 \cdot x^2+330 \cdot x+241)/13500$.

Wir erkennen die ersten drei Bezeichnungen als die allgemeine Lösung der homogenen Gleichung (sehen Sie Beispiel 1, oben). Wenn yh die Lösung zur homogenen Gleichung d.h. $y_h = K_1 \cdot e^{-3x} + K_2 \cdot e^{5x} + K_3 \cdot e^{2x}$ darstellt. Sie können prüfen, daß die restlichen Bezeichnungen in der Lösung oben gezeigt d.h. $y_p = (450 \cdot x^2+330 \cdot x+241)/13500$, setzen eine bestimmte Lösung der ODE fest

Anmerkung: Dieses Resultat ist für alle nicht-homogenen linearen ODEs, kann d.h. die Lösung der homogenen Gleichung gegeben, $y_h(x)$, die Lösung der entsprechenden nicht-homogenen Gleichung, $y(x)$, wie geschrieben werden

$$y(x) = y_h(x) + y_p(x),$$

wo $y_p(x)$ eine bestimmte Lösung der ODE ist.

Dieses überprüfen $y_p = (450 \cdot x^2+330 \cdot x+241)/13500$, ist in der Tat eine bestimmte Lösung des ODE, verwenden das folgende:

```
'd1d1d1Y(X)-4*d1d1Y(X)-11*d1Y(X)+30*Y(X) = X^2' ENTER
      'Y(X)=(450*X^2+330*X+241)/13500' ENTER
                  SUBST  EVAL
```

Geben Sie dem Rechner ungefähr 10 Sekunden, um das Resultat zu produzieren: ' X^2 = X^2 '.

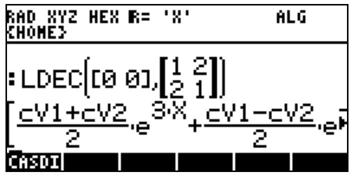
Beispiel 3 - Lösen eines Systems der linearen Differentialgleichungen mit konstanten Koeffizienten.
Betrachten Sie das System der linearen Differentialgleichungen:

$$x_1'(t) + 2x_2'(t) = 0,$$
$$2x_1'(t) + x_2'(t) = 0.$$

In der algebraischen Form wird dieses wie geschrieben :

$$\mathbf{A} \cdot \mathbf{x}'(t) = 0, \text{ wo } \mathbf{A} = \begin{bmatrix} 1 & 2 \\ 2 & 1 \end{bmatrix}.$$

Das System kann gelöst werden, indem man Funktion LDEC mit Argumenten [0.0] und Matrix A verwendet, wie im folgenden Schirm mit ALG Modus gezeigt worden:



Die Lösung wird als Vektor gegeben, der die Funktionen [x1(t) enthält,x2(t)]. Das Betätigen von von ˅ löst den Matrix-Verfasser aus, dem Benutzer erlaubend, die zwei Bestandteile des Vektors zu sehen. Um alle Details jedes Bestandteils zu sehen, betätigen Sie das EDIT! weicher Menüschlüssel. Überprüfen Sie, daß die Bestandteile sind:

Funktion DESOLVE

Der Rechner liefert Funktion DESOLVE (Differentialgleichung-Löser) um bestimmte Arten von Differentialgleichungen zu lösen. Die Funktion erfordert als Eingang die Differentialgleichung und die unbekannte Funktion, und bringt die Lösung zur Gleichung zurück, wenn vorhanden. Sie können einen Vektor auch zur Verfügung stellen, der die Differentialgleichung und die Ausgangsbedingungen, anstelle von nur einer Differentialgleichung, als Eingang zu DESOLVE enthält. Die Funktion DESOLVE ist im CALC/DIFF Menü vorhanden. Beispiele der DESOLVE Anwendungen werden unterhalb des Verwendens des RPN Modus gezeigt.

Beispiel 1 - Lösen Sie das erstrangige ODE:

$$dy/dx + x^2 \cdot y(x) = 5.$$

Im Rechnergebrauch:

```
'd1y(x)+x^2*y(x)=5' ENTER 'y(x)' ENTER DESOLVE
```

Die bereitgestellte Lösung ist { ' y = (INT(5*EXP(xt^3/3), xt, x)+C0)*1/EXP(x^3/3)} ' } d.h.

$$y(x) = \exp(-x^3/3) \cdot \left(\int 5 \cdot \exp(x^3/3) \cdot dx + cC_0 \right)$$

Das Variable ODETYPE

Sie beachten in den Weichmenu Schlüsselaufklebern neuen variablen angerufenen ODETY (ODETYPE). Die Variable wird mit dem Anruf zur DESOL Funktion produziert und eine Zeichenkette hällt, welche die Art von ODE zeigt, das als Eingang für DESOLVE verwendet wird. Betätigen Sie ODETY, um "den 1. Linearen Auftrag" zu erreichen die Zeichenkette.

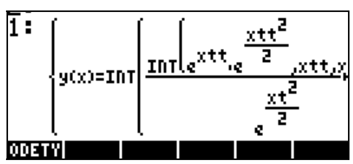
Beispiel 2 -- Lösen Sie das zweitrangige ODE:

$$d^2y/dx^2 + x\ (dy/dx) = \exp(x).$$

Im Rechnergebrauch:

'd1d1y(x)+x*d1y(x) = EXP(x)' ENTER 'y(x)' ENTER DESOLVE

Das Ergebnis ist ein Ausdruck mit zwei impliziten Integralrechnungen, und zwar,



Für diese bestimmte Gleichung jedoch stellen wir fest, daß die linke Seite der Gleichung d/dx(x dy/dx) darstellt, so das ODE jetzt geschrieben wird:

$$d/dx(x\ dy/dx) = \exp x,$$

und

$$x\ dy/dx = \exp x + C.$$

Zunächst können wir schreiben

$$dy/dx = (C + \exp x)/x = C/x + e^x/x.$$

Im Rechner können Sie versuchen zu integrieren:

‘d1y(x) = (C + EXP(x))/x’ ENTER ‘y(x)’ ENTER DESOLVE

Das Resultat ist { ‘y(x) = INT((EXP(xt)+C)/xt,xt,x)+C0’ }, d.h.,

$$y(x) = \int \cdot \frac{e^x + C}{x} dx + C_0$$

Die Integration eigenhändig durchführend, können wir sie bis zu nur erhalten:

$$y(x) = \int \cdot \frac{e^x}{x} dx + C \cdot \ln x + C_0$$

weil das Integral von exp(x)/x nicht in geschlossenem vorhanden ist, bilden Sie sich.

<u>Beispiel 3</u> - Lösen einer Gleichung mit Ausgangsbedingungen. Lösen Sie

$$d^2y/dt^2 + 5y = 2 \cos(t/2),$$

mit Ausgangsbedingungen

$$y(0) = 1.2,\ y'(0) = -0.5.$$

In the calculator, use:

[‘d1d1y(t)+5*y(t) = 2*COS(t/2)’ ‘y(0) = 6/5’ ‘d1y(0) = -1/2’] ENTER
‘y(t)’ ENTER
DESOLVE

Beachten Sie, daß die Ausgangsbedingungen zu ihren genauen Ausdrücken geändert wurden, ' y(0) = 6/5', anstatt ' y(0)=1.2 ' und ' d1y(0) = -1/2 ', anstatt, ' d1y(0) = -0,5 '. Das Ändern zu diesen genauen Ausdrücken erleichtert die Lösung.

Anmerkung: Bruchausdrücke für dezimales Wertgebrauch Funktion →Q erhalten (Sehen Sie Kapitel 5).

Die Lösung ist:

Presse EVAL EVAL to vereinfachen das Resultat zu

‘y(t) = -((19*5*SIN(√5*t)-(148*COS(√5*t)+80*COS(t/2)))/190)’.

Betätigen Sie VAR ODETY, um die Zeichenkette linear in diesem Fall zu erhalten “`Linear w/ cst coeff`” für die ODE-Art.

Laplace Wandelt Um

Das Laplace wandeln von einem Funktion f(t) produziert eine Funktion F(s) im Bildgebiet um, das verwendet werden kann, um die Lösung einer linearen Differentialgleichung zu finden, die f(t) durch algebraische Methoden mit einbezieht. Die Schritte, die in diese Anwendung mit einbezogen werden, sind drei:

1. Gebrauch des Laplace wandelt Bekehrte die lineare ODE um, die f(t) in eine algebraische Gleichung mit einbezieht.
2. Das Unbekannte F(s) wird für in das Bildgebiet durch algebraische Handhabung gelöst.
3. Ein Gegenteil Laplace wandeln wird verwendet, die Bildfunktion umzuwandeln um, die in Schritt 2 in die Lösung in das Differentialgleichung f(t) gefunden wird.

Definitionen

Das <u>Laplace wandeln</u> für die Funktion f(t) ist die Funktion F(s) definiert wie um

$$\mathrm{L}\{f(t)\} = F(s) = \int_0^\infty f(t) \cdot e^{-st}\, dt.$$

Das Bild variables s kann sein, und es ist im Allgemeinen, eine komplizierte Zahl.

Viele praktische Anwendungen von Laplace wandelt mit.einbeziehen ein ursprüngliches Funktion f(t) um, in dem t Zeit z.B. Steuersysteme in den elektrischen oder hydraulischen Stromkreisen darstellt. In den meisten Fällen ist man an der Systemantwort nach Zeit t 0 > interessiert, so wandeln die Definition des Laplace um, oben gegeben, mit.einbezieht eine Integration für Werte von t größer als null.

Das umgekehrte Laplace wandeln Diagramme die Funktion F(s) auf das ursprüngliche Funktion f(t) im Zeitgebiet d.h., $L^{-1}\{F(s)\} = f(t)$.

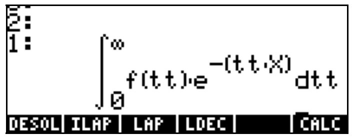
Das Windungintegral oder das Windungprodukt von zwei Funktionen f(t) und g(t), in dem g in Zeit verschoben wird, wird wie definiert

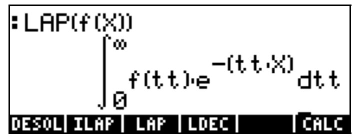
$$(f * g)(t) = \int_0^t f(u) \cdot g(t-u) \cdot du .$$

Laplace wandeln und Gegenteile im Rechner um

Der Rechner liefert die Funktionen EINHÜLLEN und ILAP, zum des Laplace zu errechnen wandeln um und das umgekehrte Laplace wandeln beziehungsweise von einem Funktion f(VX) um, in dem VX die CAS Rückstellung unabhängige Variable ist, die Sie auf ' X ' einstellen sollten. So bringt der Rechner umwandeln zurück, oder Gegenteil wandeln als Funktion von X um. Die Funktionen HÜLLEN ein und ILAP sind unter dem CALC/DIFF Menü vorhanden. Die Beispiele werden im RPN Modus ausgearbeitet, aber, sie zum ALG Modus zu übersetzen ist direkt. Zur Durchführung dieser Beispiele setzen Sie das CAS auf reellen oder exakten Modus.

Beispiel 1 - Sie können die Definition des Laplace erhalten umwandeln Gebrauch das folgende: ‘f(X)’ ENTER LAP im RPN Modus oder LAP(F(X))in ALG Modus. Der Rechner bringt das Resultat zurück (RPN, nach links; ALG, Recht):





Vergleichen Sie diese Ausdrücke mit dem, der früh in der Definition des Laplace gegeben wird, umwandeln d.h.

$$\mathsf{L}\{f(t)\} = F(s) = \int_0^\infty f(t) \cdot e^{-st} dt,$$

und Sie beachten, daß die CAS Rückstellung variables X auf dem Gleichung Verfasserschirm das variable s in dieser Definition ersetzt. Folglich wenn Sie den Funktion LAP verwenden, bekommen Sie eine Funktion von X zurück, das das Laplace umwandeln vom f(X) ist.

Beispiel 2 - Stellen Sie das Laplace umwandeln vom f(t) = $e^{2t} \cdot \sin(t)$. Gebrauch:
' EXP(2*X)*SIN(X) ' ENTER LAP der Rechner bringt das Resultat zurück:
1/(SQ(X-2)+1). Drücken Sie EVAL, um 1/(X^2-4X+5) zu erhalten.

Das Ergebnis auf Papier, würde so aussehen

$$F(s) = \mathsf{L}\{e^{2t} \cdot \sin t\} = \frac{1}{s^2 - 4 \cdot s + 5}$$

Beispiel 3 - Stellen Sie das umgekehrte Laplace umwandeln von F(s) = vom sin(s) fest. Gebrauch:
'SIN(X)' ENTER ILAP. Der Rechner bringt das Resultat zurück:
'ILAP(SIN(X))', Bedeutung, daß es kein Closed-formausdruck f(t) gibt, so daß f(t) = L^{-1}\{sin(s)\}.

Beispiel 4 - Stellen Sie das umgekehrte Laplace umwandeln von F(s) = 1/s^3. Gebrauch:'1/X^3' ENTER ILAP. Der Rechner bringt das Resultat zurück: 'X^2/2' das als $\mathsf{L}^{-1}\{1/s^3\} = t^2/2$ gedeutet wird

Beispiel 5 - Stellen Sie das Laplace umwandeln vom Funktion f(t) = cos (a·t+b) vom Lattich fest. Gebrauch: 'COS(a*X+b)' `ENTER` LAP Der Rechner bringt das Resultat zurück:

$$\frac{X}{SQ(X)+SQ(a)}\cdot COS(b)-SIN(b)\cdot\frac{a}{SQ(X)+SQ(a)}$$

Drücken Sie `EVAL` um –(a sin(b) – X cos(b))/(X^2+a^2) zu erhalten. Die Transformation wird wie folgt interpretiert: L {cos(a·t+b)} = (s·cos b – a·sin b)/(s^2+a^2).

Laplace wandeln Theoreme um

Um Ihnen zu helfen das Laplace festzustellen wandeln Sie von den Funktionen um, die Sie eine Anzahl von Theoremen verwenden können, von denen einige nachstehend aufgeführt werden. Einige Beispiele der Theoremanwendungen sind auch eingeschlossen

· Unterscheidung Theorem für die erste Ableitung. Lassen Sie f_o der Ausgangszustand für f(t) sein, d.h., f(0) = f_o, dann

$$L\{df/dt\} = s\cdot F(s) - f_o.$$

Beispiel 1 – Die Geschwindigkeit eines Partikel v(t) wird als v(t) = dr/dt definiert, , indem r = r(t) die Position des Partikels ist. Lassen Sie r_o = r(0), und R(s) =L{r(t)}, dann umwandeln der Geschwindigkeit können als V(s) = L{v(t)}=L{dr/dt}= s·R(s)-r_o. geschrieben werden.

· Unterscheidung Theorem für die zweite Ableitung. Lassen Sie f_o = f(0), und $(df/dt)_o = df/dt|_{t=0}$, dann $L\{d^2f/dt^2\} = s^2\cdot F(s) - s\cdot f_o - (df/dt)_o$.

Beispiel 2 –Als weitere Verfolgung zu Beispiel 1, wird die Beschleunigung a(t) alsa(t) = d^2r/dt^2 definiert. Wenn die Ausgangsgeschwindigkeit v_o = v(0) = $dr/dt|_{t=0}$ ist, dann wandeln das Laplace von der Beschleunigung kann wie geschrieben werden um:

$$A(s) = L\{a(t)\} = L\{d^2r/dt^2\}= s^2\cdot R(s) - s\cdot r_o - v_o.$$

· Unterscheidung Theorem für die n-th Ableitung.
Lassen Sie $f^{(k)}{}_o = d^kf/dx^k|_{t=0}$, und $f_o = f(0)$ sein, dann

$$L\{d^nf/dt^n\} = s^n\cdot F(s) - s^{n-1}\cdot f_o - \ldots - s\cdot f^{(n-2)}{}_o - f^{(n-1)}{}_o.$$

· Linearitättheorem: $L\{af(t)+bg(t)\} = a\cdot L\{f(t)\} + b\cdot L\{g(t)\}$.

· Unterscheidung Theorem für die Bildfunktion. Lassen Sie $F(s) = L\{f(t)\}$, dann $d^nF/ds^n = L\{(-t)^n\cdot f(t)\}$.

Beispiel 3 – Lassen Sie $f(t) = e^{-at}$, mit dem Rechner mit 'EXP(-a*X)' ENTER LAP, bekommen Sie '1/(X+a)', or $F(s) = 1/(s+a)$. Die dritte Ableitung dieses Ausdruckes kann errechnet werden, indem man verwendet:

'X' ENTER ↱ ∂ 'X' ENTER ↱ ∂ 'X' ENTER ↱ ∂ EVAL

Das Resultat ist
'-6/(X^4+4*a*X^3+6*a^2*X^2+4*a^3*X+a^4)', or
$d^3F/ds^3 = -6/(s^4+4\cdot a\cdot s^3+6\cdot a^2\cdot s^2+4\cdot a^3\cdot s+a^4)$.

- Integration Theorem. Lassen Sie $F(s) = L\{f(t)\}$ sein, dann

$$L\left\{\int_0^t f(u)du\right\} = \frac{1}{s}\cdot F(s).$$

- Windungtheorem. Lassen Sie $F(s) = L\{f(t)\}$ und $G(s) = L\{g(t)\}$ sein, dann

$$L\left\{\int_0^t f(u)g(t-u)du\right\} = L\{(f*g)(t)\} =$$

$$L\{f(t)\}\cdot L\{g(t)\} = F(s)\cdot G(s)$$

Beispiel 4 – Mit dem Windungtheorem finden Sie das Laplace , von umzuwandeln (f*g)(t), wenn f(t) = sin(t), und g(t) = exp(t). Um F(s) = L{f(t)} zu finden, und G(s) = L{g(t)}, benützen Sie: 'SIN(X)' ENTER LAP EVAL . Resultat, '1/(X^2+1)', i.e., F(s) = 1/(s^2+1).
Auch, 'EXP(X)' ENTER LAP Resultat, '1/(X-1)', d.h., G(s) = 1/(s-1). Also, L{(f*g)(t)} = F(s)·G(s) = 1/(s^2+1)·1/(s-1) = 1/((s-1)(s^2+1)) = 1/(s^3-s^2+s-1).

- Schiebetheorem für eine Verschiebung nach rechts. Lassen Sie F(s) = L{f(t)}, dann L{f(t-a)}=e^{-as}·L{f(t)} = e^{-as}·F(s).
- Schiebetheorem für eine Verschiebung nach links. Lassen Sie F(s) = L{f(t)}, und a >0, dann

$$L\{f(t+a)\} = e^{as} \cdot \left(F(s) - \int_0^a f(t) \cdot e^{-st} \cdot dt \right).$$

- Ähnlichkeittheorem. Lassen Sie F(s) = L{f(t)}, und a>0, dann L{f(a·t)} = (1/a)·F(s/a).
- Dämpfung Theorem. Lassen Sie F(s) = L{f(t)}, dann L{e–bt·f(t)} = F(s+b).
- Abteilung Theorem. Lassen Sie F(s) = L{f(t)}, dann

$$L\left\{\frac{f(t)}{t}\right\} = \int_s^\infty F(u)du.$$

- Laplace wandeln von einer periodischen Funktion von Periode T um:

$$L\{f(t)\} = \frac{1}{1-e^{-sT}} \cdot \int_0^T f(t) \cdot e^{-st} \cdot dt.$$

- Begrenzungstheorem für den Ausgangswert. Lassen Sie F(s) = L{f(t)} dann

$$f_0 = \lim_{t \to 0} f(t) = \lim_{s \to \infty}[s \cdot F(s)].$$

- Begrenzungstheorem für den abschließenden Wert: Lassen Sie F(s) = L{f(t)}, dann

$$f_\infty = \lim_{t\to\infty} f(t) = \lim_{s\to 0}[s \cdot F(s)].$$

Funktion Dreieck Diracs und Schrittfunktion Heavisides

In der Analyse der Steuersysteme ist- es üblich, eine Art Funktionen, die bestimmtes körperliches Auftreten wie die plötzliche Aktivierung eines Schalters (Funktion, H(t) Schritt Heavisides) oder des plötzlichen darstellen, blitzschnell, Spitze in einem Eingang zum System (Funktion, δ(t) Dreieck Diracs) zu verwenden. Diese gehören einer Kategorie Funktionen, die bekannt sind als die generalisierten oder symbolischen Funktionen [z.B., sehen Sie Friedman, B., 1956, Grundregeln und Techniken von angewandter Mathematics, Dover Publications Inc., neues York (Neuauflage 1990)].

Die formale Definition der <u>Funktion Dreieck Diracs</u>, δ(x), is δ(x) = 0, for x ≠0, and

$$\int_{-\infty}^{\infty} \delta(x)dx = 1.0.$$

Auch wenn f(x) eine stetige Dauerfunktion ist, dann

$$\int_{-\infty}^{\infty} f(x)\delta(x - x_0)dx = f(x_0).$$

Eine Deutung für das Integral oben, paraphrasiert von Friedman (1990), ist, Daß die δ-Funktion den Wert des Funktion "auswählt f(x) an x = x_0. Funktion Dreieck Diracs wird gewöhnlich durch einen aufwärts Pfeil am Punkt x = x0, Funktion Dreieck Diracs wird gewöhnlich durch einen aufwärts Pfeil am Punkt x_0 hat.

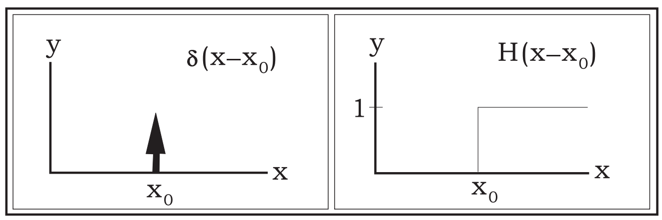
Schrittfunktion Heavisides, H(x), wird wie definiert

$$H(x) = \begin{cases} 1, & x > 0 \\ 0, & x < 0 \end{cases}$$

Auch für ein stetige Dauerfunktion f(x),

$$\int_{-\infty}^{\infty} f(x)H(x-x_0)dx = \int_{x_0}^{\infty} f(x)dx.$$

Funktion Dreieck Diracs und Schrittfunktion Heavisides werden durch dH/dx = δ(x) bezogen. Die zwei Funktionen werden in der Abbildung unten veranschaulicht.



Sie können prüfen das $L\{H(t)\} = 1/s$,
von welchem es folgt, dass $L\{U_o \cdot H(t)\} = U_o/s$,
wo U_o eine Konstante ist. Auch, $L^{-1}\{1/s\}=H(t)$,
und $L^{-1}\{ U_o /s\}= U_o \cdot H(t)$.

Auch mit einem Schiebetheorem für eine Verschiebung rechts, $L\{f(t\text{-}a)\}=e^{-as} \cdot L\{f(t)\} = e^{-as} \cdot F(s)$, können wir schreiben$L\{H(t\text{-}k)\}=e^{-ks} \cdot L\{H(t)\} = e^{-ks} \cdot (1/s) = (1/s) \cdot e^{-ks}$.

Ein anderes wichtiges Resultat, bekannt, als das <u>zweite Schiebetheorem für eine Verschiebung rechts</u>, das ist- $L^{-1}\{e^{-as} \cdot F(s)\}=f(t\text{-}a) \cdot H(t\text{-}a)$, mit $F(s) = L\{f(t)\}$.

Im Rechner gekennzeichnet die Heaviside Schrittfunktion H(t) einfach als ' 1 '. Umwandeln im Rechnergebrauch überprüfen: 1 ENTER LAP. Resultat ist '1/X', d.h., L{1} = 1/s. Ähnlich, 'U0' ENTER LAP , produziert das Resultat 'U0/X', d.h., $L\{U_0\} = U_0/s$.

Sie können Funktion Dreieck Diracs im Rechner erhalten, indem Sie verwenden: 1 ENTER ILAP

Das Resultat ist `'Delta(X)'`.

Dieses Resultat ist einfach symbolisch, d.h. können Sie nicht einen numerischen Wert für, ein Sagen ' Delta(5 finden) '.

Diesem Resultat kann definiert werden das Laplace umwandeln für Funktion Dreieck Diracs, weil von $L^{-1}\{1.0\} = \delta(t)$, resultiert in $L\{\delta(t)\} = 1.0$

Auch mit dem Schiebetheorem für eine Verschiebung rechts, $L\{f(t-a)\}=e^{-as}\cdot L\{f(t)\} = e^{-as}\cdot F(s)$, können wir schreiben $L\{\delta(t-k)\}=e^{-ks}\cdot L\{\delta(t)\} = e^{-ks}\cdot 1.0 = e^{-ks}$.

Anwendungen von Laplace wandeln in der Lösung der linearen Oden um

Am Anfang des Abschnitts auf Laplace wandelt uns anzeigte um, daß Sie diese verwenden konnten umwandeln, um eine lineare ODE im Zeitgebiet in eine algebraische Gleichung im Bildgebiet umzuwandeln. Die resultierende Gleichung wird dann für eine Funktion F(s) durch algebraische Methoden gelöst, und die Lösung zur ODE wird, indem man das umgekehrte Laplace verwendet, umwandeln auf F(s) gefunden.
Die Theoreme auf Ableitungen einer Funktion d.h.

$$L\{df/dt\} = s\cdot F(s) - f_o,$$

$$L\{d^2f/dt^2\} = s^2\cdot F(s) - s\cdot f_o - (df/dt)_o,$$

und, im allgemeinem,

$$L\{d^nf/dt^n\} = s^n\cdot F(s) - s^{n-1}\cdot f_o - \ldots - s\cdot f^{(n-2)}{}_o - f^{(n-1)}{}_o,$$

seien Sie besonders nützlich, wenn Sie ein ODE in eine algebraische Gleichung umwandeln.

Beispiel 1 - Die erste Auftrag Gleichung lösen,

$$dh/dt + k\cdot h(t) = a\cdot e^{-t},$$

indem wir verwenden, wandelt Laplace, wir kann schreiben um:

$$L\{dh/dt + k\cdot h(t)\} = L\{a\cdot e^{-t}\},$$

$$L\{dh/dt\} + k\cdot L\{h(t)\} = a\cdot L\{e^{-t}\}.$$

Anmerkung: 'EXP(-X)' `ENTER` LAP , produziert '1/(X+1)', d.h., $L\{e^{-t}\} = 1/(s+1)$.

Mit $H(s) = L\{h(t)\}$, und $L\{dh/dt\} = s\cdot H(s) - h_o$, wo $h_o = h(0)$, ist die umgewandelte Gleichung $s\cdot H(s)-h_o+k\cdot H(s) = a/(s+1)$.

Benutzen Sie den Rechner, um für H(s), durch Schreiben zu lösen:

'X*H-h0+k*H=a/(X+1)' `ENTER` 'H' ISOL

Das Resultat ist 'H=((X+1)*h0+a)/(X^2+(k+1)*X+k)'.

Um die Lösung zur ODE zu finden, h(t), müssen wir das umgekehrte Laplace verwenden umwandeln, wie folgt:

OBJ→ `◄` `◄` `EVAL` Isoliert rechte Seite des letzten Ausdruckes

ILAP Erreicht das umgekehrte Laplace umwandeln.

Das Ergebnis lautet $\dfrac{a\cdot e^{k\cdot X}+((k-1)\cdot ho-a)\cdot e^{X}}{(k-1)\cdot e^{X}\cdot e^{k\cdot X}}$.

Ersetzen Sie in diesem Ausdruck X mit t und vereinfachen Sie diesen, ist das Ergebnis

$$h(t) = a/(k\text{-}1)\cdot e^{-t} + ((k\text{-}1)\cdot h_o\text{-}a)/(k\text{-}1)\cdot e^{-kt}.$$

Überprüfen Sie, was die Lösung zur ODE sein würde, wenn Sie die Funktion LDEC verwenden:

'a*EXP(-X)' `ENTER` 'X+k' `ENTER` LDEC `EVAL`

Die Lösung lautet: $\frac{a\cdot e^{k\cdot X}+((k-1)\cdot cC0-a)\cdot e^{X}}{(k-1)\cdot e^{X}\cdot e^{k\cdot X}}$ d.h.

$$h(t) = a/(k\text{-}1)\cdot e^{-t} +((k\text{-}1)\cdot cC_o\text{-}a)/(k\text{-}1)\cdot e^{-kt}.$$

So stellt cC0 in den Resultaten von LDEC den Ausgangszustand h(0) dar.

Anmerkung: Wenn Sie die Funktion LDEC benützen um ein lineares ODE von Auftrag n im f(X) zu lösen, wird das Resultat in n Konstanten cC0, cC1, cC2 ausgedrückt, …, gegeben cC(n-1), die Ausgangsbedingungen darstellend f(0), f'(0), f"(0), …, $f^{(n-1)}$ (0).

Beispiel 2 - Gebrauch Laplace wandelt um, um die zweitrangige lineare Gleichung zu lösen,

$$d^2y/dt^2+2y = \sin 3t.$$

Das Verwenden von Laplace wandelt, wir kann schreiben um

$$L\{d^2y/dt^2+2y\} = L\{\sin 3t\},$$

$$L\{d^2y/dt^2\} + 2\cdot L\{y(t)\} = L\{\sin 3t\}.$$

Anmerkung: 'SIN(3*X)' ENTER LAP EVAL produziert '3/(X^2+9)', d.h., $L\{\sin 3t\}=3/(s^2+9)$.

Mit $Y(s) = L\{y(t)\}$, und $L\{d^2y/dt^2\} = s^2\cdot Y(s) - s\cdot y_o - y_1$, wo $y_o = h(0)$ und $y_1 = h'(0)$ sind, ist die veränderte Vergleichung

$$s^2\cdot Y(s) - s\cdot y_o - y_1 + 2\cdot Y(s) = 3/(s^2+9).$$

Benutzen Sie den Rechner, um für Y(s), durch Schreiben zu lösen:

'X^2*Y-X*y0-y1+2*Y=3/(X^2+9)' ENTER 'Y' ISOL

Das Resultat ist

‘Y=((X^2+9)*y1+(y0*X^3+9*y0*X+3))/(X^4+11*X^2+18)’.

Um die Lösung zum ODE zu finden, y(t), müssen wir das umgekehrte Laplace verwenden umwandeln, wie folgt:

OBJ→ [◄] [◄] Isoliert rechte Seite des letzten Ausdruckes
ILAP [EVAL] Erreicht das umgekehrte Laplace umwandeln

Resultat:

$$\frac{(7\cdot\sqrt{2}\cdot y1+3\cdot\sqrt{2})\cdot SIN(\sqrt{2}\cdot X)+14\cdot y0\cdot COS(\sqrt{2}\cdot X)-2\cdot SIN(3\cdot X)}{14}$$

d.h.,

$$y(t) = -(1/7) \sin 3x + y_o \cos \sqrt{2}x + (\sqrt{2}\ (7y_1+3)/14) \sin \sqrt{2}x.$$

Überprüfen Sie, was die Lösung zum ODE sein würde, wenn Sie die Funktion LDEC verwenden:

‘SIN(3*X)’ [ENTER] ‘X^2+2’ [ENTER] LDEC [EVAL]

Das Resultat:

$$\frac{(7\cdot\sqrt{2}\cdot cC1+3\cdot\sqrt{2})\cdot SIN(\sqrt{2}\cdot X)+14\cdot cC0\cdot COS(\sqrt{2}\cdot X)-2\cdot SIN(3\cdot X)}{14}$$

d.h. dasselbe wie vorher mit cC0 = y0 und cC1 = y1.

Anmerkung: Mit den zwei Beispielen können wir bestätigen, was wir vorher anzeigten daß Funktion ILAP gebrauch Laplace transformiert und Gegenteil umwandelt um lineares ODEs zu lösen, welches die rechte Seite der Gleichung und der charakteristischen Gleichung des entsprechenden homogenen ODE gegeben wird.

Beispiel 3 - Betrachten Sie die Gleichung

$$d^2y/dt^2+y = \delta(t-3),$$

wo δ(t) Dirac's delta Funktion ist.

Das Verwenden von von Laplace wandelt, wir kann schreiben um:

$$L\{d^2y/dt^2+y\} = L\{\delta(t\text{-}3)\},$$

$$L\{d^2y/dt^2\} + L\{y(t)\} = L\{\delta(t\text{-}3)\}.$$

Mit '`Delta(X-3)`' ENTER LAP , erzeugt der Rechner EXP(-3*X),
i.e., L{?(t-3)} = e^{-3s}.

Mit Y(s) = L{y(t)}, und $L\{d^2y/dt^2\} = s^2\cdot Y(s) - s\cdot y_o - y_1$, wo y_o = h(0) und y_1 = h'(0), ist die verwanderte Vergleichung $s^2\cdot Y(s) - s\cdot y_o - y_1 + Y(s) = e^{-3s}$. Benutzen Sie den Rechner, um für Y(s), durch Schreiben zu lösen:

'X^2*Y-X*y0-y1+Y=EXP(-3*X)' ENTER 'Y' ISOL

Resultat: 'Y=(X*y0+(y1+EXP(-(3*X))))/(X^2+1)'.

Um die Lösung zum ODE zu finden, y(t), müssen wir das umgekehrte Laplace verwenden umwandeln, wie folgt :

OBJ→ ◄ ◄	Isoliert die rechte Seite des letzten Ausdruckes
ILAP EVAL	Erreicht das umgekehrte Laplace umwandeln

Das Resultat ist 'y1*SIN(X)+y0*COS(X)+SIN(X-3)*Heaviside(X-3)'.

Anmerkungen:

[1]. Eine Alternative die umgekehrte Laplace Transformation des Ausdruckes '(X*y0+(y1+EXP(-(3*X))))/(X^2+1)' zu erhalten, ist den Ausdruck in Teilfraktionen zu zerlegen, d.h.

'y0*X/(X^2+1) + y1/(X^2+1) + EXP(-3*X)/(X^2+1)',
und verwenden Sie das Linearitättheorem des umgekehrten Laplace umwandeln

$$L^{-1}\{a\cdot F(s)+b\cdot G(s)\} = a\cdot L^{-1}\{F(s)\} + b\cdot L^{-1}\{G(s)\},$$

Damit wir schreiben

$$L^{-1}\{y_o\cdot s/(s^2+1)+y_1/(s^2+1)) + e^{-3s}/(s^2+1)) \} =$$

$$y_o\cdot L^{-1}\{s/(s^2+1)\}+ y_1\cdot L^{-1}\{1/(s^2+1)\}+ L^{-1}\{e^{-3s}/(s^2+1))\},$$

Dann benutzen wir den Rechner, um das folgende zu erreichen:

'X/(X^2+1)' ENTER ILAP Resultat, 'COS(X)', d.h., $L^{-1}\{s/(s^2+1)\}$= cos t.
'1/(X^2+1)' ENTER ILAP Resultat, 'SIN(X)', d.h., $L^{-1}\{1/(s^2+1)\}$= sin t.
'EXP(-3*X)/(X^2+1)' ENTER ILAP Resultat, SIN(X-3)*Heaviside(X-3)'

[2]. Das letzte Resultat ist 'ILAP(EXP(-3*X)/(X^2+1))', d.h., der Rechner kann nicht das umgekehrte Laplace finden, von dieser Bezeichnung umzuwandeln. In diesem Fall jedoch können wir das symbolische Resultat (an zweiter Stelle verschiebentheorem für eine Verschiebung rechts) verwenden

$$L^{-1}\{e^{-as}\cdot F(s)\}=f(t-a)\cdot H(t-a),$$

wenn wir ein umgekehrtes Laplace finden können, für 1/(s2+1) umzuwandeln. Mit dem Rechner Versuch '1/(X^2+1)' ENTER ILAP. Das resultiert in '0.+1*SIN(X)', oder, mit EVAL 'SIN(X)'. Also, $L^{-1}\{e^{-3s}/(s^2+1))\}$ = sin(t-3)·H(t-3) soll die Lösung zum ursprünglichen ODE wie geschriebe wenn wir ein umgekehrtes Laplace finden können, für 1/(s2+1) umzuwandeln. Mit dem Rechner Versuch '1/(X^2+1)' ENTER ILAP. Das resultiert in '0.+1*SIN(X)', oder, mit EVAL 'SIN(X)'. Also, $L^{-1}\{e^{-3s}/(s^2+1))\}$ = sin(t-3)·H(t-3) soll die Lösung zum ursprünglichen ODE wie geschrieben werden:

$$y(t) = y_o \cos t + y_1 \sin t + \sin(t-3)\cdot H(t-3).$$

Überprüfen Sie, was die Lösung zum ODE sein würde, wenn Sie die Funktion LDEC verwenden:

'Delta(t-3)' ENTER 'X^2+1' ENTER LDEC EVAL

Das Resultat ist:

'SIN(X-3)*Heaviside(X-3) + cC1*SIN(X) + cC0*COS(X)+'.

Beachten Sie bitte, daß das variable X in diesem Ausdruck wirklich das variable t im ursprünglichen ODE darstellt und daß die Variable $$t in diesem Ausdruck eine Platzhaltervariable ist.

So kann die Übersetzung der Lösung im Papier wie geschrieben werden:

$$y(t) = Co \cdot \cos t + C_1 \cdot \sin t + \sin(t-3) \cdot H(t-3)$$

Wenn wir dieses Resultat mit dem vorhergehenden Resultat für y(t) vergleichen, stellen wir daß fest

$cC_o = y_o$, $cC_1 = y_1$,

Das Definieren und das Verwenden von von Schritt Heavisides arbeiten im Rechner

Das vorhergehende Beispiel versah etwas Erfahrung mit dem Gebrauch von Funktion Dreieck Diracs als Eingang zu einem System (d.h., in der rechten Seite des Ods das System beschreiben). In diesem Beispiel möchten wir Schrittfunktion Heavisides, H(t) verwenden. Im Rechner können wir diese Funktion wie definieren:

'H(X) = IFTE(X>0, 1, 0)' ENTER ← DEF

Diese Definition verursacht das variable H im weichen Menüschlüssel des Rechners.

Beispiel 1 – Einen Plot von H(t-2) z.B. zu sehen benutzen eine FUNCTION Art Plot (sehen Sie Kapitel 12):

- Presse ← 2D/3D , gleichzeitig im RPN Modus, zum Zugang zum PLOT SETUP Fenster.
- Verändere `TYPE` in `FUNKTION`, wann nötig

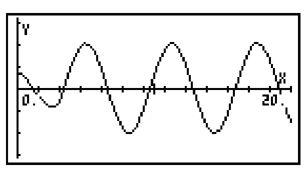
- Verändere EQ in 'H(X-2)'.
- Seien Sie sicher daß `Indep` ist eingestellt auf 'X'.
- Presse NXT OK um zur normalen Rechneranzeige zurückzugehen .
- Presse ← WIN , gleichzeitig, um das PLOT-Fenster zugänglich zu machen.
- Verändere die H-Ansicht Strecke von 0 bis 20 und die V-VIEL von -2 bis 2.
- Betätigen Sie ERASE DRAW, um die Funktion zu plotten .

Gebrauch von der Funktion H(X) mit LDEC, LAP oder ILAP, wird nicht im Rechner erlaubt. Sie müssen die Hauptresultate verwenden, die früh beim Beschäftigen die Heaviside Schrittfunktion d.h. bereitgestellt werden
$L\{H(t)\} = 1/s$, $L^{-1}\{1/s\}=H(t)$,
$L\{H(t-k)\}=e^{-ks}\cdot L\{H(t)\} = e^{-ks}\cdot(1/s) = \cdot(1/s)\cdot e^{-ks}$ and $L^{-1}\{e^{-as}\cdot F(s)\}=f(t-a)\cdot H(t-a)$.

Beispiel 2 – Die Funktion $H(t-t_o)$, wann multipliziert zu einer Funktion f(t), d.h., $H(t-t_o)f(t)$, hat den Effekt die Funktion
f(t) at $t = t_o$ ein zu schalten. Zum Beispiel, die Lösung bekommen in Beispiel 3, hier oben, war $y(t) = y_o \cos t + y_1 \sin t + \sin(t-3)\cdot H(t-3$ Nehmen Sie uns verwenden das Initiale Zustände $y_o = 0.5$, and $y_1 = -0.25$ an. Zu sehen lassen Sie uns diese Funktion plotten, um, was es wie aussieht:

- Presse ← 2D/3D , gleichzeitig als ob in RPN Modus, um das PLOT SETUP-Fenster zugänglich zu machen
- Verändere `TYPE` in `FUNKTION`, wann nötig
- Verändere EQ in '0.5*COS(X)-0.25*SIN(X)+SIN(X-3)*H(X-3)'.
- Seien Sie sicher daß `Indep` auf 'X' ist eingestellt.
- Betätigen Sie ERASE DRAW, um die Funktion zu plotten .
- Presse EDIT NXT LABEL, um das Plot zu sehen .

Das resultierende Diagramm sieht wie dieses aus:



Beachten Sie, daß das Signal mit einem verhältnismäßig kleinen Umfang beginnt, aber plötzlich, bei t=3, schaltet es zu einem Schwingungssignal mit einem größeren Umfang. Der Unterschied zwischen dem Verhalten des Signals vor und nach t = 3 ist die "Schaltung auf" der bestimmten Lösung $y_p(t) = \sin(t-3)\cdot H(t-3)$. Das Verhalten des Signals vor t = 3 stellt den Beitrag der homogenen Lösung dar, $y_h(t) = y_o \cos t + y_1 \sin t$.

Die Lösung einer Gleichung mit einem treibenden Signal, das durch eine Heaviside Schrittfunktion gegeben wird, wird unten gezeigt.

<u>Beispiel 3</u> – Stellen Sie die Lösung zur Gleichung fest, $d^2y/dt^2+y = H(t-3)$, wo H(t) ist Schrittfunktion Heavisides. Das Verwenden von Laplace wandelt, können wir umschreiben als: $L\{d^2y/dt^2+y\} = L\{H(t-3)\}$, $L\{d^2y/dt^2\} + L\{y(t)\} = L\{H(t-3)\}$. Die letzte Bezeichnung in diesem Ausdruck ist: $L\{H(t-3)\} = (1/s)\cdot e^{-3s}$. Mit $Y(s) = L\{y(t)\}$, und $L\{d^2y/dt^2\} = s^2\cdot Y(s) - s\cdot y_o - y_1$, wo $y_o = h(0)$ und $y_1 = h'(0)$ ist, ist die umgewandelte Gleichung:
$s^2\cdot Y(s) - s\cdot y_o - y_1 + Y(s) = (1/s)\cdot e^{-3s}$. Ändern Sie, falls notwendig, den CAS-Modus auf exakt.Benutzen Sie den Rechner, um für Y(s), durch Schreiben zu lösen:

‘X^2*Y-X*y0-y1+Y=(1/X)*EXP(-3*X)’ ENTER ‘Y’ ISOL

Das Resultat ist

‘Y=(X^2*y0+X*y1+EXP(-3*X))/(X^3+X)’.

Um die Lösung zum ODE zu finden, y(t), müssen wir die umgekehrte Laplace Umwandlung verwenden, als folgt:

OBJ→ ◄ ◄	Isoliert die rechte Seite des letzten Ausdruckes
ILAP	Erreicht die umgekehrte Laplace Umwandlung

Das Resultat ist:

‘y1*SIN(X-1)+y0*COS(X-1)-(COS(X-3)-1)*Heaviside(X-3)’.

Somit haben wir als Lösung: $y(t) = y_o \cos t + y_1 \sin t + H(t-3)\cdot(1+\sin(t-3))$.

Überprüfen Sie die Lösung der ODE, unter Benutzung der Funktion LDEC:

'H(X-3)' `ENTER` [ENTER] 'X^2+1' `ENTER` LDEC

Die Lösung lautet:

$$\mathrm{SIN}(X)\cdot\int_0^{+\infty}\frac{\mathrm{IFTE}(ttt-3>0,1,0)}{e^{X\cdot ttt}}\,dttt\ +cC1\cdot\mathrm{SIN}(X)+cC0\cdot\mathrm{COS}(X)$$

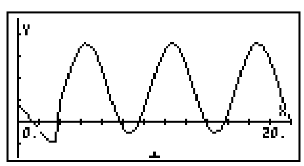
Beachten Sie, dass in diesem Ausdruck die Variable X eigentlich die Variable t in der ursprünglichen ODE darstellt und die Variable ttt in diesem Ausdruck nur eine Scheinvariable darstellt. Somit könnte die Übertragung der Lösung auf Papier so niedergeschrieben werden:

$$y(t) = Co\cdot\cos t + C_1\cdot\sin t + \sin t\cdot\int_0^{\infty} H(u-3)\cdot e^{-ut}\cdot du.$$

<u>Beispiel 4</u> -- Plotten Sie die Lösung zu Beispiel 3 mit den gleichen Werten des y_o und y_1 die im Plot von Beispiel 1 oben verwendet werden. Wir plotten jetzt die Funktion

y(t) = 0.5 cos t –0.25 sin t + (1+sin(t-3))·H(t-3).

In der Strecke 0 < sollte t < 20 und Ändern der vertikalen Strecke zu (-1.3), das Diagramm wie dieses aussehen:



Wieder gibt es einen neuen Bestandteil zur Bewegung, die an t=3 nämlich das bestimmte Lösung yp(t) = geschaltet wird [1+sin(t-3)]?H(t-3), das die Natur der Lösung für t>3 ändert.

Die Heaviside Schrittfunktion kann mit einer konstanten Funktion und mit linearen Funktionen kombiniert werden, um das Quadrat zu erzeugen, dreieckig und sah begrenzte Impulse des Zahnes, wie folgt:

- Quadratischer Impuls der Größe Uo im Abstand $a < t < b$:

$$f(t) = Uo[H(t\text{-}a)\text{-}H(t\text{-}b)].$$

- Dreieckiger Impuls mit einem Maximalwert Uo, erhöhend von $a < t < b$ und verringern sich von $b < t < c$:

$$f(t) = U_o\cdot ((t\text{-}a)/(b\text{-}a)\cdot[H(t\text{-}a)\text{-}H(t\text{-}b)]+(1\text{-}(t\text{-}b)/(b\text{-}c))[H(t\text{-}b)\text{-}H(t\text{-}c)]).$$

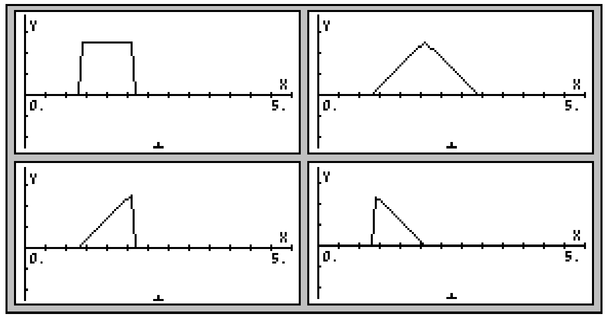
- Sägezahnimpuls, der auf einen Maximalwert Uo für $a < t < b$, plötzlich unten fallend auf null an $t = b$ sich erhöht ab:

$$f(t) = U_o\cdot (t\text{-}a)/(b\text{-}a)\cdot[H(t\text{-}a)\text{-}H(t\text{-}b)].$$

- Sägezahnimpuls, der sich plötzlich auf ein Maximum von Uo an $t = a$, dann linear verringernd bis null für $a < t < b$ erhöht

$$f(t) = U_o\cdot[1\text{-}(t\text{-}a)/(b\text{-}1)]\cdot[H(t\text{-}a)\text{-}H(t\text{-}b)].$$

Beispiele der Plots, die durch diese Funktionen, für Uo = 1, a = 2, b = 3, c = 4, Xstrecke = erzeugt werden (0.5) und Ystrecke = (-1, 1.5), werden in den Abbildungen unten gezeigt:



Fourier-Reihe

Fourier-Reihen sind die Reihen, die den Sinus und Kosinusfunktionen gewöhnlich verwenden, um periodische Funktionen mit einzubeziehen. Eine Funktion f(x) soll periodisch sein, von Periode T, wenn f(x+T) = f(t) ist. Z.B. weil sin(x+2π) = Sünde x und cos(x+2π) = cos x, die Funktionen sin und cos sind periodische Funktionen 2π. Wenn die zwei Funktionen f(x) und g(x) von Periode T periodisch sind, dann ist ihre lineare Kombination h(x) = a·f(x) + b·g(x), auch periodisch für Periode T. Eine T-periodische Funktion f(t) kann in einer Reihe von Sinus und Kosinusfunktionen erweitert werden, die als Fourier-Reihe bekannt sind, die vorbei gegeben wirden von

$$f(t) = a_0 + \sum_{n=1}^{\infty}\left(a_n \cdot \cos\frac{2n\pi}{T}t + b_n \cdot \sin\frac{2n\pi}{T}t\right)$$

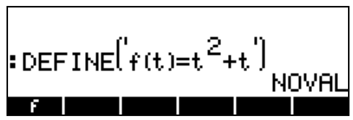
wo die Koëffizienten a_n und b_n gegeben werden

$$a_0 = \frac{1}{T}\int_{-T/2}^{T/2} f(t)\cdot dt, \quad a_n = \frac{2}{T}\int_{-T/2}^{T/2} f(t)\cdot\cos\frac{2n\pi}{T}t\cdot dt,$$

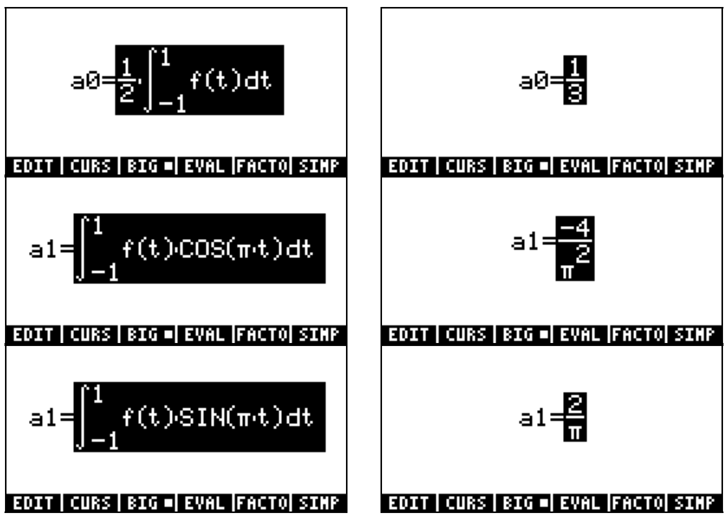
$$b_n = \int_{-T/2}^{T/2} f(t)\cdot\sin\frac{2n\pi}{T}t\cdot dt.$$

Die nachfolgenden Übungen sind im ALG-Modus, während das CAS Modus auf exakt eingestellt ist. (Wenn Sie einen Graphen erstellen, wird der CAS-

Modus auf Approx zurückgestellt. Stellen Sie sicher, dass Sie nach Erstellen des Graphs in den Exact-Modus zurückwechseln.) Nehmen Sie z.B. an daß die Funktion f(t) = daß t^2+t mit Periode T = 2 periodisch ist. Um die Koeffizienten a_0, a_1, und b_1 für die entsprechende Fourier-Reihe festzustellen, gehen wir folgendermaßen vor: Zuerst definieren Sie Funktion f(t) = t^2+t :



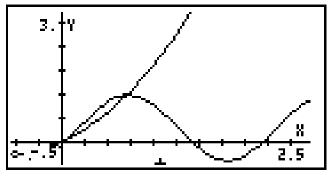
Zunächst benützen wir die Gleichung Verfasser, womit wir die Koeffizienten errechnen:



So sind die ersten drei Bezeichnungen der Funktion:

$$f(t) \approx 1/3 - (4/\pi^2)\cdot\cos(\pi\cdot t)+(2/\pi)\cdot\sin(\pi\cdot t).$$

Ein graphischer Vergleich der ursprünglichen Funktion mit der Fourier Expansion, die diese drei verwendet, zeicht, daß die Befestigung für t < 1 annehmbar ist, oder annähert. Aber dann wieder vereinbarten wir daß T/2 = 1. Folglich ist die Befestigung nur zwischen?1 < t < 1 gültig.



Die FOURIER Funktion

Eine alternative Weise, eine Fourier-Reihe zu definieren ist, indem Sie komplizierte Zahlen verwenden, wie folgt:

$$f(t) = \sum_{n=-\infty}^{+\infty} c_n \cdot \exp(\frac{2in\pi t}{T}),$$

wo

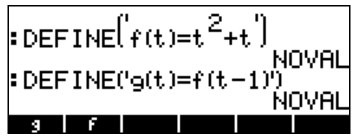
$$c_n = \frac{1}{T}\int_0^T f(t)\cdot \exp(\frac{2\cdot i\cdot n\cdot \pi}{T}\cdot t)\cdot dt, \quad n = -\infty,...,-2,-1,0,1,2,...\infty.$$

Die FOURIER Funktion liefert das Koeffizient c_n der Kompliziertform der Fourier-Reihe, die die Funktion f(t) und den Wert von n gegeben wird. Die Funktion FOURIER erfordert Sie, den Wert der Periode (T) einer T-periodischen Funktion zu speichern in die CAS variable PERIODE, bevor sie die Funktion benennt. Die Funktion FOURIER ist im DERIV Unterprogramm innerhalb des CALC Menüs (CALC).vorhanden.

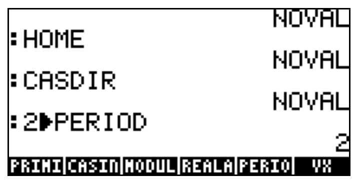
Fourier-Reihe für eine quadratische Funktion

Stellen Sie die Koeffizienten c_0, c_1, und c_2 für die Funktion $f(t) = t^2+t$, mit Periode T = 2 fest. (Anmerkung: Weil das Integral, daß von Funktion FOURIER verwendet wird, im Abstand [0,T] errechnet wird, während das, daß vorher definiert wurde, im Abstand [- T/2, T/2] errechnet wurde, müssen wir die Funktion in den T-axis durch subtrahieren des T/2 von t verschieben, d.h. wir benutzen $g(t) = f(t-1) = (t-1)^2+(t-1)$.)

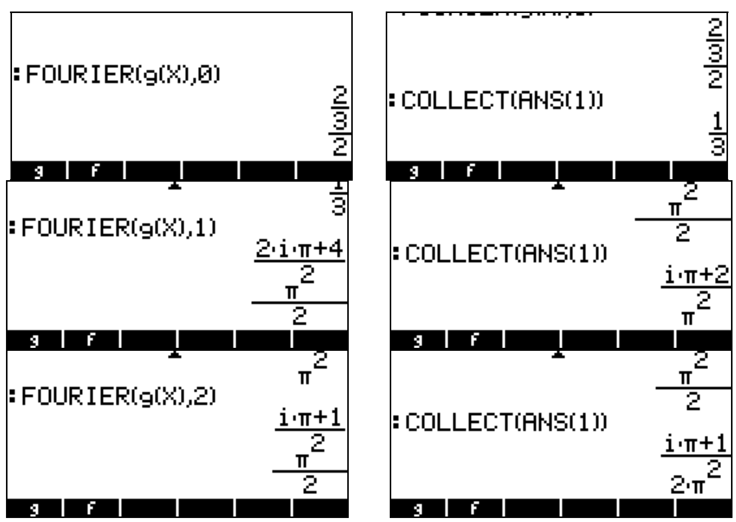
Mit dem Rechner im ALG Modus, definieren wir zuerst Funktionen f(t) und g(t):



Zunächst ziehen wir auf das CASDIR Unterverzeichnis unter HOME um, um den Wert der variablen PERIODE zu ändern z.B. ., ← (hold) UPDIR ENTER VAR CASDI ENTER 2 STO▶ PERIO ENTER



Gehen Sie zum Unterverzeichnis zurück, in dem Sie Funktionen f und g definierten, und errechnen Sie die Koeffizienten (Falls verlangt, akzeptieren Sie den Wechsel in den Complex-Modus):

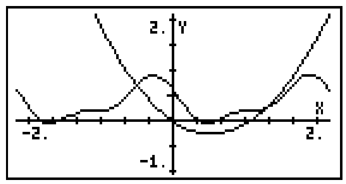


Also, $c_0 = 1/3$, $c_1 = (\pi \cdot i+2)/\pi^2$, $c_2 = (\pi \cdot i+1)/(2\pi^2)$.

Die Fourier-Reihe mit drei Elementen wird geschrieben als

$$g(t) \approx \mathrm{Re}[(1/3) + (\pi \cdot i+2)/\pi^2 \cdot \exp(i \cdot \pi \cdot t)+ (\pi \cdot i+1)/(2\pi^2) \cdot \exp(2 \cdot i \cdot \pi \cdot t)].$$

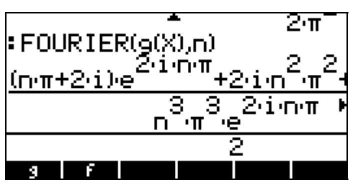
Ein Plot des verschobenen Funktion g(t) und der Fourier-Reihe der hineinpasst, folgt:

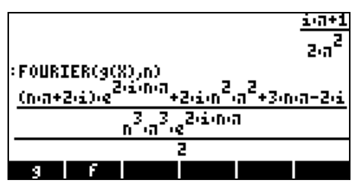


Die Befestigung ist für annehmbar für 0<t<2, obgleich nicht so gut wie im vorhergehenden Beispiel.

Ein allgemeiner Ausdruck für cn

Die Funktion FOURIER kann einen allgemeinen Ausdruck für das Koeffizient c_n der Expansion der komplizierten Fourier-Reihe zur Verfügung stellen.. Z.B. mit dem gleichen Funktion g(t) wie vorher, wird die allgemeine Bezeichnung c_n (Abbildungen zeigen normalen Schriftkegel und kleine Schriftkegelanzeigen) gegeben durch:



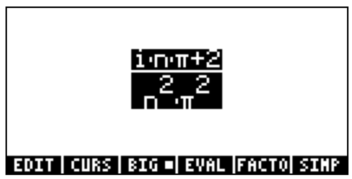


Der allgemeine Ausdruck ist geprüft worden von:

$$c_n = \frac{(n\pi + 2i) \cdot e^{2in\pi} + 2i^2 n^2 \pi^2 + 3n\pi - 2i}{2n^3 \pi^3 \cdot e^{2in\pi}}$$

Wir können diesen Ausdruck sogar weiter vereinfachen, indem wir Euler's Formel für komplizierte Zahlen verwenden, nämlich $e^{2in\pi}$ = cos(2nπ) + i·sin(2nπ) = 1 + i·0 = 1, weil cos(2nπ) = 1, and sin(2nπ) = 0 für n Ganzzahl ist.

Mit dem Rechner können Sie den Ausdruck im Gleichung Verfasser vereinfachen (↱ EQW) indem Sie $e^{2in\pi}$ = 1 ersetzen. Die Abbildung zeigt den Ausdruck nach Vereinfachung:

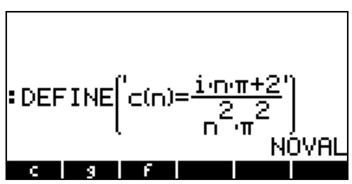


Das Resultat ist: $c_n = (i\cdot n\cdot\pi+2)/(n^2\cdot\pi^2)$.

Die komplizierte Fourier-Reihe zusammenfügen

Den allgemeinen Ausdruck für c_n feststellend, können wir eine begrenzte komplizierte Fourier-Reihe zusammenfügen, indem wir die Summierungfunktion (Σ) im Rechner verwenden wie folgt:

- Zuerst definieren Sie eine Funktion c(n), welche das allgemeine Bezeichnung c_n in der komplizierten Fourier-Reihe darstellt.
-



- Zunächst definieren Sie die begrenzte komplizierte Fourier-Reihe, F(X, k), wo X die unabhängige Variable ist und k stellt die Zahl der zu verwenden Bezeichnungen fest. Tadellos möchten wir diese begrenzte komplizierte Fourier-Reihe schreiben als

$$F(X,k) = \sum_{n=-k}^{k} c(n) \cdot \exp(\frac{2 \cdot i \cdot \pi \cdot n}{T} \cdot X)$$

Jedoch weil die Funktion c(n) nicht für n = 0 definiert wird, ist es uns geraten, den Ausdruck zu schreiben als

$$F(X,k,c0) = c0 +$$

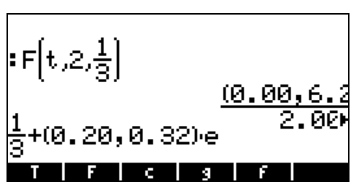
$$\sum_{n=1}^{k} [c(n) \cdot \exp(\frac{2 \cdot i \cdot \pi \cdot n}{T} \cdot X) + c(-n) \cdot \exp(-\frac{2 \cdot i \cdot \pi \cdot n}{T} \cdot X)],$$

Oder, in der Rechnereintragung Linie als:

DEFINE('F(X,k,c0) = c0+Σ(n=1,k,c(n)*EXP(2*i*π*n*X/T)+
c(-n)*EXP(-(2*i*π*n*X/T))'),

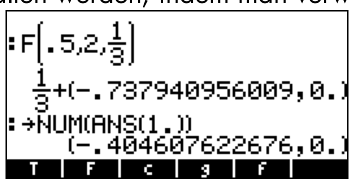
wo T die Periode ist, T = 2. Die folgenden Abbildungen zeigen die Definition der Funktion F und die Speicherung von T = 2

Die Funktion kann verwendet werden, um den Ausdruck für die komplizierte Fourier-Reihe für einen begrenzten Wert von k zu erzeugen. Z.B. für k = 2, c_0 = 1/3 und t verwendent als die unabhängige Variable, können wir F(t, 2, 1/3) auswerten, um zu erhalten:



Dieses Resultat zeigt nur die erste Bezeichnung (c0) und Teil der ersten exponentialen Bezeichnung in der Reihe. Das Format der Ziffernanzeige-einheit wurde zur Verlegenheit mit 3 Dezimalstrichen geändert, um in der Lage zu sein einige der Koeffizienten in der Expansion und im Exponenten zu zeigen. Wie erwartet, sind die Koeffizienten komplizierte Zahlen.

Die Funktion F, auf dieser Weise definiert, ist prima für das Erhalten von Werten der begrenzten Fourier-Reihe. Z.B. ein einzelner Wert der Reihe z.B. F(0.5,2,1/3), kann erhalten werden, indem man verwendet:

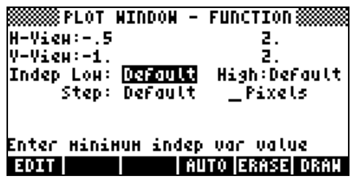


Änderung annehmen in der Approx Modus, wenn daß verlangt wird Das Resultat ist der Wert
–0.40467…. Der tatsächliche Wert der Funktion g(0.5) ist g(0.5) = -0,25. Die folgenden Berechnungen zeigen, wie gut die Fourier-Reihe diesen Wert approximiert, während die Zahl der Bestandteilen in der Reihe, gegeben durch k, sich erhöht:

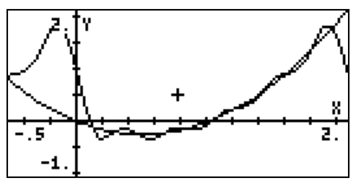
F (0.5, 1, 1/3) = (-0.303286439037,0.)
F (0.5, 2, 1/3) = (-0.404607622676,0.)
F (0.5, 3, 1/3) = (-0.192401031886,0.)
F (0.5, 4, 1/3) = (-0.167070735979,0.)
F (0.5, 5, 1/3) = (-0.294394690453,0.)
F (0.5, 6, 1/3) = (-0.305652599743,0.)

Um die Resultate von der Reihe mit denen der ursprünglichen Funktion zu vergleichen, laden Sie diese Funktionen in den PLOT – FUNKTION Eingang Form (Y= , gleichzeitig, wenn RPN Modus verwendet wird):

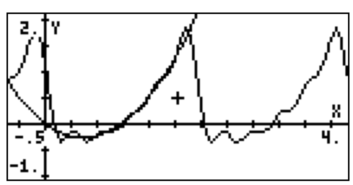
Ändern Sie die Begrenzungen auf das Plot-Fenster (WIN) wie folgt:



Betätigen Sie die Weichmenü Schlüssel ERASE DRAW um den Plot zu produzieren:



Beachten Sie daß die Reihe, mit 5 Bezeichnungen, das Diagramm der Funktion sehr nah im Abstand 0 bis 2 "umarmt" (d.h., durch die Periode T = 2). Sie können auch eine Periodizität im Diagramm der Reihe beachten. Diese Periodizität ist einfach sichtbar zu machen, indem man die X-Strecke des Plots erweitert (-0,5.4):



Fourier-Reihe für eine dreieckige Welle

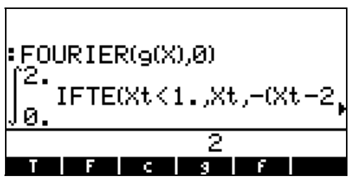
Betrachten Sie die Funktion

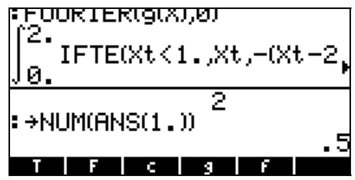
$$g(x) = \begin{cases} x, & if\ 0 < x < 1 \\ 2 - x, & if\ 1 < x < 2 \end{cases}$$

welches wir annehmen periodisch zu sein, mit Periode T = 2. Diese Funktion kann im Rechner, im ALG Modus, durch den Ausdruck definiert werden

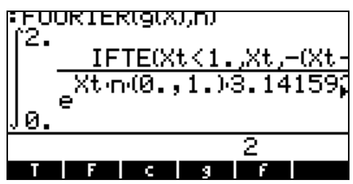
DEFINE('g(X) = IFTE(X<1,X,2-X)')

Wenn Sie dieses Beispiel begannen, nachdem Sie Beispiel 1 beendet hatten, haben Sie bereits einen Wert von 2 gespeichert in der CAS Variable PERIODE. Wenn Sie nicht sicher sind, überprüfen Sie den Wert dieser Variable, und speichern Sie 2 in ihm, wenn sie vielleicht benötigt wird. Der Koeffizient c_0 für die Fourier-Reihe wird wie folgendes errechnet:





Der Rechner bittet um eine Änderung am Approx Modus wegen der Integration der Funktion IFTE(), die im Integrand umfaßt wird. Annehmend, produziert die Änderung im Approx Modus c_0 = 0,5. Wenn wir jetzt einen generischen Ausdruck für den Koeffizient c_n erhalten möchten, benützen wir:



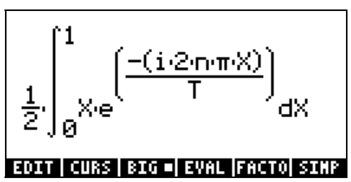
Der Rechner bringt ein Integral zurück, das nicht numerisch ausgewertet werden kann, weil er vom Parameter n abhängt. Der Koeffizient kann noch errechnet werden, indem wir seine Definition im Rechner schreiben, d.h.

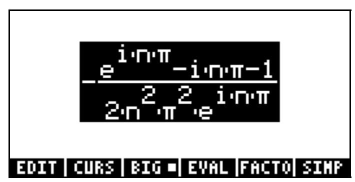
$$\frac{1}{2}\cdot\int_0^1 X\cdot EXP\left(-\frac{i\cdot 2\cdot n\cdot\pi\cdot X}{T}\right)\cdot dX +$$

$$\frac{1}{2}\cdot\int_1^2 (2-X)\cdot EXP\left(-\frac{i\cdot 2\cdot n\cdot\pi\cdot X}{T}\right)\cdot dX$$

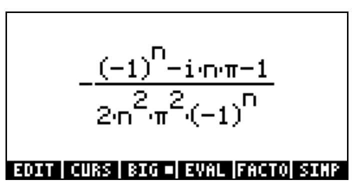
wo T = 2 die Periode ist. Der Wert von T kann gespeichert werden, wenn wir verwenden:

Geben Sie die erste der oben dargestellten Integrale in den EquationWriter ein und wählen Sie den gesamten Ausdruck aus, erhalten Sie unter Verwendung von EVAL nachfolgendes Ergebnis:

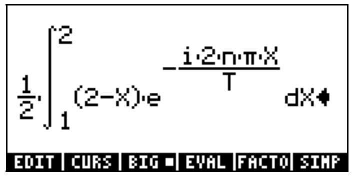
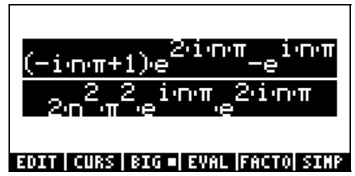




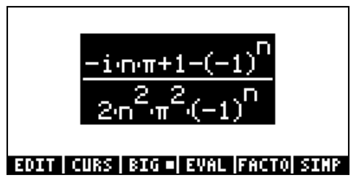
Erinnern Sie sich $e^{in\pi} = \cos(n\pi) + i\cdot\sin(n\pi) = (-1)^n$. Diesen Ersatz im Resultat oben durchführend, haben wir:



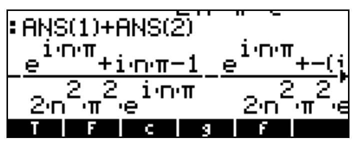
Betätigen Sie ENTER ENTER um dieses Resultat zum Schirm zu kopieren. Dann reaktivieren Sie den Gleichung Verfasser, um das zweite Integral zu errechnen, welches das Koeffizient c_n definiert, nämlich,

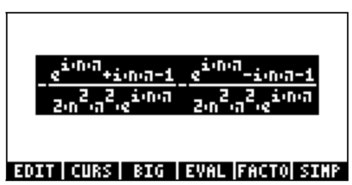
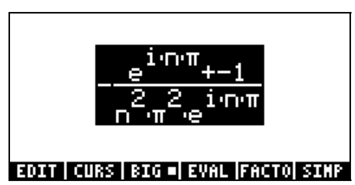
Noch einmal $e^{in\pi} = (-1)^n$ ersetzend, und $e^{2in\pi} = 1$ benützend, erhalten wir:



Betätigen Sie ENTER ENTER um dieses zweite Resultat zum Schirm zu kopieren. Jetzt fügen Sie ANS(1) und ANS(2) zu, um den vollen Ausdruck für c_n zu erhalten:



Durch drücken der Taste ▼ wird das Ergebnis im EquationWriter ausgegeben, wo dieses wie folgt vereinfacht werden (SIMP) kann:

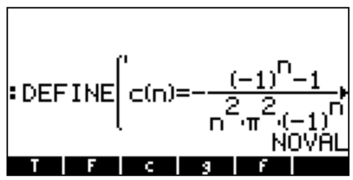
 

Nochmals $e^{in\pi} = (-1)^n$ ersetzen resultiert in

Dieses Resultat wird verwendet, um das Funktion c(n) wie folgt zu definieren:

DEFINE('c(n) = - (((-1)^n-1)/(n^2*π^2*(-1)^n)')

d.h.,

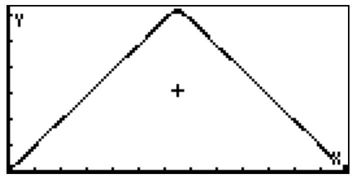


Zunächst definieren wir Funktion F(X, k, c0), um die Fourier-Reihe zu errechnen (wenn Sie Beispiel 1 durchführten, haben Sie diese Funktion bereits gespeichert):

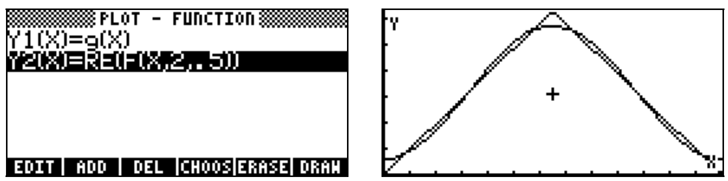
DEFINE('F(X,k,c0) = c0+Σ(n=1,k,c(n)*EXP(2*i*π*n*X/T)+
c(-n)*EXP(-(2*i*π*n*X/T))'),

Um die ursprüngliche Funktion und die Fourier-Reihe zu vergleichen, können wir den simultanen Plot beider Funktionen produzieren. Die Details sind denen von Beispiel 1 ähnlich, außer daß wir hier eine Strecke x von 0 bis 2 und für y von 0 bis 1 benutzen, und justieren die Gleichungen zum plotten, wie hier gezeigt:

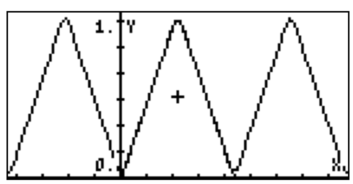
Das resultierende Diagramm wird unten für k = 5 gezeigt (die Zahl der Elementen in der Reihe ist 2k+1 11 d.h., in diesem Fall):



Vom Plot aus ist es sehr schwierig, die ursprüngliche Funktion vom Fourier-Reihe des Näherungswertens zu unterscheiden. K = 2 oder 5 Bezeichnungen in der Reihe benützend, zeigt nicht so eine gute Befestigung:



Die Fourier-Reihe kann verwendet werden um eine periodische dreieckige Welle (oder Sägezahnwelle) zu erzeugen, indem man die X-axis-Strecke z.B. von -2 bis 4 ändert. Das Diagramm hier unten gezeigt benützt k = 5:



Fourier-Reihe für eine quadratische Welle

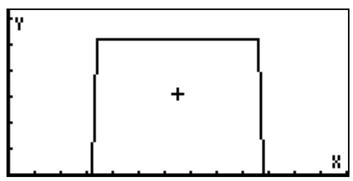
Eine quadratische Welle kann erzeugt werden, indem man folgende Funktion verwendet

$$g(x) = \begin{cases} 0, & if\ 0 < x < 1 \\ 1, & if\ 1 < x < 3 \\ 0, & if\ 3 < x < 4 \end{cases}$$

In diesem Fall die Periode T ist 4. Überprüfen Sie, den Wert des variablen T zu ändern bis 4 (use: 4 STO▶ T ENTER). Funktion g(X) kann im Rechner definiert werden, indem man verwendet

DEFINE('g(X) = IFTE((X>1) AND (X<3),1,0)')

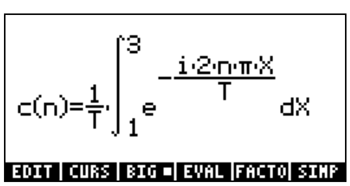
Die Funktion plottete wie folgt (horizontaler Bereich: 0 bis 4, vertikaler Bereich: 0 bis 1,2):

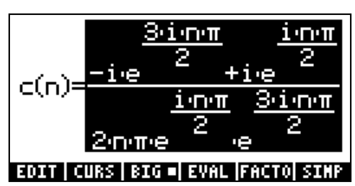


Mit einem Verfahren, daß dem der dreieckigen Form in Beispiel 2 oben ähnlich ist, können Sie herausfinden daß

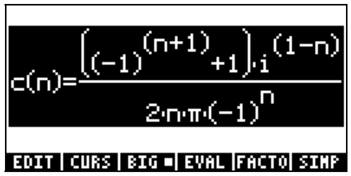
$$c_0 = \frac{1}{T} \cdot \left(\int_1^3 1 \cdot dX \right) = 0.5\,,$$

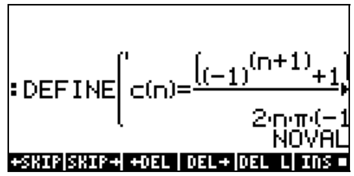
und



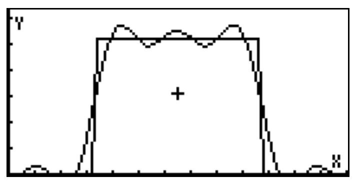


Wir können diesen Ausdruck vereinfachen, indem wir $e^{in\pi/2} = i^n$ $e^{3in\pi/2} = (-i)^n$ verwenden, um nächtstes zu erhalten:

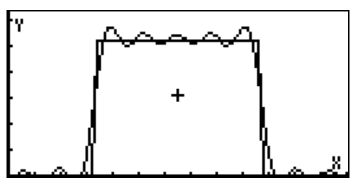




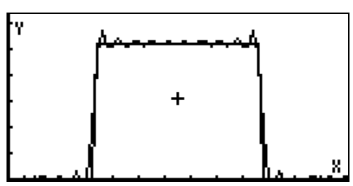
Die Vereinfachung der rechten Seite von c(n), hier oben, ist auf Papier einfacher zu machen (d.h., eigenhändig). Dann tippen Sie den Ausdruck für c(n) neu ein, wie in der Abbildung links oben gezeigt wird, um Funktion c(n) zu definieren. Die Fourier-Reihe wird mit F(X, k, c0), als in Beispielen 1 und 2 oben, mit c0 = 0,5 errechnet. Z.B. für k = 5 d.h. mit 11 Bestandteilen wird der Näherungswert unten gezeigt:



Ein besserer Näherungswert wird erreicht, indem man k = 10 verwendet, d.h.



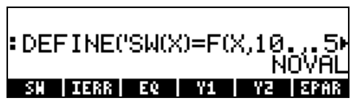
Die Befestigung für k = 20, ist sogar besser, aber es nimmt länger, um das Diagramm zu produzieren:



Anwendungen in Differenzialgleichungen in Fourier-Reihe

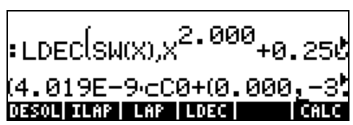
Nehmen Sie an wir möchten die periodische quadratische Welle, definiert im vorhergehenden Beispiel benützen wie die Anstrengung eines undampfierten Waagen dessen homogene Gleichung ist: $d^2y/dX^2 + 0.25y = 0$.

Wir können die Erregungskraft erzeugen, indem wir einen Näherungswert mit k = 10 aus der Fourier-Reihe heraus erreichen, indem wir SW(X) = F(X, 10, 0.5) verwenden:

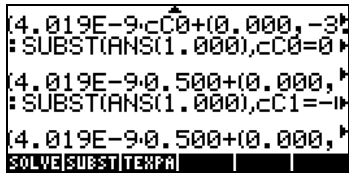


Wir können dieses Resultat als der erste Eingang zur Funktion LDEC verwenden, wenn sie benützt wird, eine Lösung zum $d^2y/dX^2 + 0.25y = SW(X)$ zu erreichen, in dem SW(X) steht für quadratische Welle Funktion von X. Der zweite Eingangseinzelteil ist die charakteristische Gleichung, die dem homogenen ODE entspricht, das oben gezeigt wird, d.h.'X^2+0.25' .

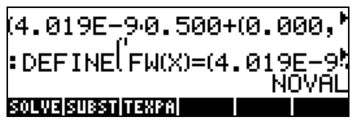
Mit diesen zwei Eingängen produziert Funktion LDEC das folgende Resultat: (das dezimale Format änderte zur Verlegenheit mit 3 Dezimalstrichen).



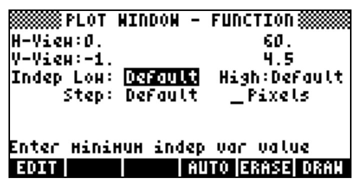
Das Betätigen von ▼ erlaubt Ihnen, den gesamten Ausdruck im Gleichung Verfasser zu sehen. Das Erforschen der Gleichung im Gleichung Verfasser deckt das Bestehen von zwei Integrationskonstanten auf, cC0 und cC1. Diese Werte würden mit Ausgangsbedingungen errechnet. Nehmen wir an daß wir die Werte cC0 = 0,5 und cC1 = -0,5 verwenden, wir können jene Werte in der Lösung oben ersetzen, indem wir Funktion SUBST verwenden (sehen Sie Kapitel 5). Verwenden Sie für diesen Fall SUBST(ANS(1), cC0=0.5) ENTER, gefolgt von SUBST(ANS(1), cC1=-0.5) ENTER. Zurück in normale Rechneranzeige können wir verwenden:



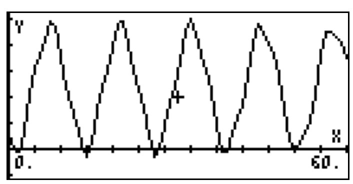
Das letzte Resultat kann als Funktion FW(X) definiert werden, wie folgt (Ausschnitt und Kleben des letzten Resultats in den Befehl):



Wir können das reale Teil dieser Funktion jetzt plotten.   Ändern Sie den Dezimalmodus auf Standard und verwenden Sie folgendes:



Die Lösung wird unten gezeigt:



Fourier-Umwandlungen

Bevor dem Darstellen des Konzeptes von Fourier-Umwandlungen, werden wir die allgemeine Definition einer Integralumwandlung besprechen.   Im allgemeinen, ein Integralumwandlung ist eine, die eine Funktion f(t) auf einer neuen Funktion F(s) bezieht, durch eine Integration der Form:

$$F(s) = \int_a^b \kappa(s,t) \cdot f(t) \cdot dt.$$

Die Funktion κ(s, t) ist bekannt als der Kern der Umwandlung.

Der Gebrauch von einem Integralumwandlung erlaubt uns, eine Funktion in ein gegebenes Spektrum der Bestandteile zu beheben. Um das Konzept eines Spektrums zu verstehen, betrachten Sie die Fourier-Reihe

$$f(t) = a_0 + \sum_{n=1}^{\infty} \left(a_n \cdot \cos \omega_n x + b_n \cdot \sin \omega_n x \right),$$

eine periodischen Funktion mit einer Periode T darstellend. Diese Fourier-Reihe kann neu geschrieben werden wie

$$f(x) = a_0 + \sum_{n=1}^{\infty} A_n \cdot \cos(\varpi_n x + \phi_n), \text{ wo}$$

$$A_n = \sqrt{a_n^2 + b_n^2}, \quad \phi_n = \tan^{-1}\left(\frac{b_n}{a_n}\right),$$

für n = 1.2,…

Die Umfänge A_n werden gekennzeichnet als das Spektrum der Funktion und werden ein Maß sein der Größe des Bestandteils von f(x) mit Frequenz f_n = n/T. Die grundlegende oder Grundfrequenz in der Fourier-Reihe ist $f_0 = 1/T$, also sind alle weiteren Frequenzen Mehrfachverbindungsstellen dieser grundlegenden Frequenz, d.h. $f_n = n \cdot f_0$. Auch wir können eine eckige Frequenz definieren, $\omega_n = 2n\pi/T = 2\pi \cdot f_n = 2\pi \cdot n \cdot f_0 = n \cdot \omega_0$, wo ω_0 die grundlegende oder grundlegende eckige Frequenz der Fourier-Reihe ist.

Mit der eckigen Frequenzdarstellung wird die Expansion der Fourier-Reihe geschrieben wie

$$f(x) = a_0 + \sum_{n=1}^{\infty} A_n \cdot \cos(\omega_n x + \phi_n).$$

$$= a_0 + \sum_{n=1}^{\infty} \left(a_n \cdot \cos \omega_n x + b_n \cdot \sin \omega_n x \right)$$

Ein Plot der Werte, A_n gegen ω_n ist die typische Darstellung eines getrennten Spektrums für eine Funktion. Das getrennte Spektrum zeigt, daß die Funktion Bestandteile am eckigen Frequenzen ω_n hat, welche ganzzahlige Vielfachen der grundlegenden eckigen Frequenz ω_0 sind.

Nehmen Sie an, daß wir mit der Notwendigkeit, eine nicht-periodische Funktion in Sinus- und Kosinusbestandteile zu erweitern gegenübergestellt werden. Eine nicht-periodische Funktion kann gehalten werden eine unendlich große Periode zu haben. So daß für einen sehr großen Wert von T, die grundlegende eckige Frequenz $\omega_0 = 2\pi/T$ eine sehr kleine Quantität wird, lasst uns sagen $\Delta\omega$. Auch die eckigen Frequenzen, die to $\omega_n = n \cdot \omega_0 = n \cdot \Delta\omega$, ($n = 1, 2, \ldots, \infty$) entsprechen, nehmen jetzt Werte näher und näher an einander an und die Notwendigkeit an einem ununterbrochenen Spektrum von Werten wird vorgeschlagen.

Die nicht-periodische Funktion kann, folglich, geschrieben werden wie

$$f(x) = \int_0^{\infty} [C(\omega) \cdot \cos(\omega \cdot x) + S(\omega) \cdot \sin(\omega \cdot x)] d\omega,$$

wo

$$C(\omega) = \frac{1}{2\pi} \cdot \int_{-\infty}^{\infty} f(x) \cdot \cos(\omega \cdot x) \cdot dx,$$

und

$$S(\omega) = \frac{1}{2\pi} \cdot \int_{-\infty}^{\infty} f(x) \cdot \sin(\omega \cdot x) \cdot dx.$$

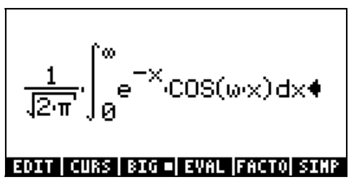
Das ununterbrochene Spektrum wird gegeben durch

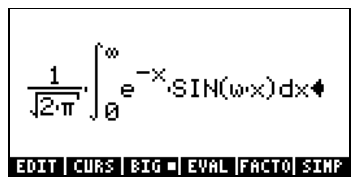
$$A(\omega) = \sqrt{[C(\omega)]^2 + [S(\omega)]^2}$$

Die Funktionen C(ω), S(ω), und A(ω) sind stetige Dauerfunktionen einer Variable ω, welches wird zu Umwandelnvariable,für den Fourier unten definiert.

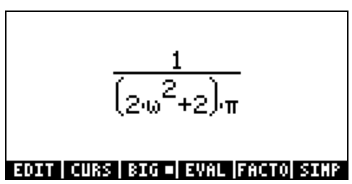
<u>Beispiel 1</u> -- Stellen Sie die Koeffizienten C(ω), S(ω), und das ununterbrochene Spektrum A(ω) fest, für Funktionen f(x) = exp(-x) für x > 0 und f(x) = 0, x < 0.

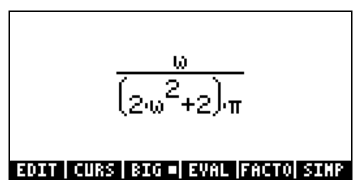
Im Rechner stellen Sie die folgenden Integrale auf und werten sie aus, um beziehungsweise C(ω)und S(ω) zu errechnen. CAS Modi sind zu Genau und Reell gesetzt.



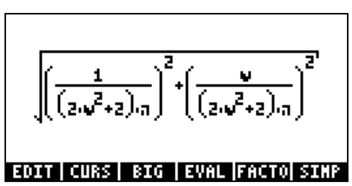


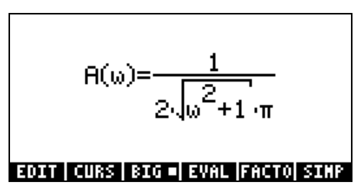
Ihre Resultate sind:



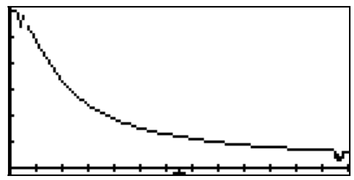


Das ununterbrochene Spektrum A(ω) wird errechnet:





Definieren Sie diesen Ausdruck als eine Funktion, indem Sie Funktion DEFINE (⬅ DEF) verwenden. Dann plotten Sie das ununterbrochene Spektrum, im Bereich 0 < ω < 10:



Definition der Fourier Umwandlungen

Unterschiedliche Arten von Fourierumwandlungen können definiert werden. Die folgenden sind die Definitionen des Sinus, im Bereich 0 < ω < 10

Fourier Sinusumwandlung

$$\mathrm{F}s\{f(t)\} = F(\omega) = \frac{2}{\pi} \cdot \int_0^\infty f(t) \cdot \sin(\omega \cdot t) \cdot dt$$

Umgekehrte Sinusumwandlung

$$\mathrm{F}_s^{-1}\{F(\omega)\} = f(t) = \int_0^\infty F(\omega) \cdot \sin(\omega \cdot t) \cdot dt$$

Fourier Kosinusumwandlung

$$\mathrm{F}c\{f(t)\} = F(\omega) = \frac{2}{\pi} \cdot \int_0^\infty f(t) \cdot \cos(\omega \cdot t) \cdot dt$$

Umgekehrte Kosinusumwandlung

$$\mathrm{F}_c^{-1}\{F(\omega)\} = f(t) = \int_0^\infty F(\omega) \cdot \cos(\omega \cdot t) \cdot dt$$

Fourier Umwandlung (korrekt)

$$\mathsf{F}\{f(t)\} = F(\omega) = \frac{1}{\sqrt{2\pi}} \cdot \int_{-\infty}^{\infty} f(t) \cdot e^{-i\omega t} \cdot dt$$

Umgekehrte Fourier Umwandlung (korrekt)

$$\mathsf{F}^{-1}\{F(\omega)\} = f(t) = \frac{1}{\sqrt{2\pi}} \cdot \int_{-\infty}^{\infty} F(\omega) \cdot e^{-i\omega t} \cdot dt$$

Beispiel 1 -- Stellen Sie die Fourier-Transformation fest der Funktion f(t) = exp(-t), für t > 0 und f(t) = 0, für t<0.

Das ununterbrochene Spektrum F(ω) wird errechnet mit dem Integral:

$$\frac{1}{\sqrt{2\pi}} \int_0^{\infty} e^{-(1+i\omega)t} dt = \lim_{\varepsilon \to \infty} \frac{1}{\sqrt{2\pi}} \int_0^{\varepsilon} e^{-(1+i\omega)t} dt$$

$$= \lim_{\varepsilon \to \infty} \frac{1}{\sqrt{2\pi}} \left[\frac{1 - \exp(-(1+i\omega)t)}{1+i\omega} \right] = \frac{1}{\sqrt{2\pi}} \cdot \frac{1}{1+i\omega}.$$

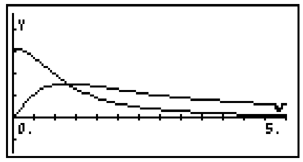
Dieses Resultat kann rationalisiert werden, indem man Zähler und Nenner mit dem Paronym des Nenners multipliziert, nämlich 1-iω. Jetzt ist das Resultat:

$$F(\omega) = \frac{1}{\sqrt{2\pi}} \cdot \frac{1}{1+i\omega} = \frac{1}{\sqrt{2\pi}} \cdot \left(\frac{1}{1+i\omega} \right) \cdot \left(\frac{1-i\omega}{1-i\omega} \right)$$

$$= \frac{1}{\sqrt{2\pi}} \left(\frac{1}{1+\omega^2} - i \cdot \frac{\omega}{1+\omega^2} \right)$$

welches eine komplizierte Funktion ist.

Die realen und Imaginärteile der Funktion können geplottet werden, wie unten gezeigt wird



Anmerkungen:
Der Absolutwert der Fourier-Transformation, $|F(\omega)|$, ist das Frequenzspektrum der ursprünglichen Funktionen f(t). Für das Beispiel oben gezeigt, $|F(\omega)| = 1/[2\pi(1+\omega^2)]^{1/2}$. Der Plot von $|F(\omega)|$ gegen ω wurde vorher schon gezeigt.

Einige Funktionen, wie konstante Werte, Sin x, exp(x), x2, etc., haben keine Fourier-Transformation. Funktionen, die bis null schnell genug gehen während x zur Unbegrenztheit geht, haben Fourierumwandlungen.

Eigenschaften der Fourierumwandlungen

Linearitäten: Wenn a and b Konstanten, und f und g Funktionen sind, dann $F\{a \cdot f + b \cdot g\} = a\ F\{f\} + b\ F\{g\}$.

Umwandlung der teilweisen Ableitungen. Lassen Sie u = u(x, t). Wenn die Fourierumwandlung das Variable x transformiert, dann

$$F\{\partial u/\partial x\} = i\omega\ F\{u\},\quad F\{\partial^2 u/\partial x^2\} = -\omega^2\ F\{u\},$$
$$F\{\partial u/\partial t\} = \partial F\{u\}/\partial t,\ F\{\partial^2 u/\partial t^2\} = \partial^2 F\{u\}/\partial t^2$$

Windung: Für Anwendungen der Fourierumwandlungen wird der Betrieb der Windung definiert wie

$$(f * g)(x) = \frac{1}{\sqrt{2\pi}} \cdot \int f(x-\xi) \cdot g(\xi) \cdot d\xi.$$

Die folgenden Eigenschaft wird beeinflüsst für Windung:

$$F\{f*g\} = F\{f\} \cdot F\{g\}.$$

Fast Fourier Transform (FFT)

Die schnelle Fourierumwandlung / Fast Fourier Transform ist ein Computeralgorithmus, durch den man eine getrennte Fourierumwandlung / Discrete Fourier Transorm (DFT) sehr leistungsfähig errechnen kann. Dieser Algorithmus hat Anwendungen in der Analyse der unterschiedlichen Arten von zeitabhängigen Signale, von Turbulenzmaße zu Kommunikation Signale.

Die getrennte Fourierumwandlung einer Reihenfolge der $\{x_j\}$, j = 0, 1, 2, …, n-1, ist eine neue begrenzte Reihenfolge $\{X_k\}$, definiert wie

$$X_k = \frac{1}{n}\sum_{j=0}^{n-1} x_j \cdot \exp(-i \cdot 2\pi kj / n), \qquad k = 0,1,2,\ldots,n-1$$

Die direkte Berechnung der Reihenfolge X_k bezieht n^2 Produkte, die enormen Mengen Zeit des Computers (oder Rechners) miteinbeziehen würden, besonders für große Werte von n. Die schnelle Fourier-Umwandlung verringert die Zahl der Arbeiten vom Auftrag von $n \cdot \log_2 n$. Z.B. für n = 100, erfordert das FFT ungefähr 664 Bearbeitungen, während die direkte Berechnung 10.000 Bearbeitungen erfordern würde. So wird die Zahl den Betrieben, die das FFT verwenden, durch einen Faktor von 10000/664 verringert ≈ 15.

Das FFT arbeitet an die Reihenfolge $\{x_j\}$ indem es sie in eine Anzahl von kürzeren Reihenfolgen verteilt. Die DFT's der kürzeren Reihenfolgen werden errechnet und später zusammen kombiniert in einer in hohem Grade leistungsfähigen Weise. Für Details über den Algorithmus verweisen wir Sie z.B. auf Newland, D.E., 1993, "Eine Einleitung auf gelegentliche Erschütterungen, spektrale u. Wavelet-Analyse -- Dritte Ausgabe," Longman Wissenschaftlich und Technisch, New York (Kapitel 12).

Die einzige Anforderung für die Anwendung des FFT ist, daß die Zahl n eine Energie von 2 ist, d.h. wählen Sie Ihre Daten vor, damit sie 2, 4, 8, 16, 32, 62 usw. Punkte enthält.

Beispiele der FFT Anwendungen

FFT Anwendungen beziehen normalerweise die Daten mit ein, die von einem zeitabhängigen Signal ausgeschlossen sind. Diese Daten können im Rechner eingetragen werden für die Verarbeitung, sagen wir von einem Computer oder einen Datenlogger. Oder, Sie können Ihre eigenen Daten erzeugen, indem Sie eine Funktion programmieren und einige gelegentliche Zahlen hinzufügen.

Beispiel 1 -- Definieren Sie die Funktion f(x) = 2 sin (3x) + 5 cos(5x) + 0.5*RAND., wo RAND der konstante Generator der gelegentlichen Zahl ist, der vom Rechner bereitgestellt wird. Erzeugen Sie 128 Datenpunkte, indem Sie Werte von x im Abstand (0.12.8) verwenden. Speichern Sie jene Werte in einer Reihe (array), und führen Sie ein FFT auf der Reihe aus.

Zuerst definieren wir die Funktion f(x), wie ein RPN Programm:

<< →x '2*SIN(3*x) + 5*COS(5*x)' EVAL RAND 5 * + →NUM >>

und speichern wir dieses Programm in der Variable f. Zunächst schreiben wir das folgende Programm, um 2^m Datenwerte zwischen a und b zu erzeugen. Das Programm nimmt die Werte von m, von a und von b:

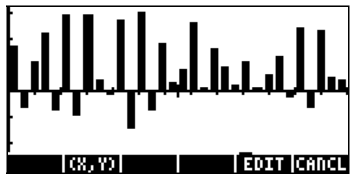
<< → m a b << '2^m' EVAL → n << '(b-a)/(n+1)' EVAL → Dx << 1 n FOR j 'a+(j-1)*Dx' EVAL f NEXT n →ARRY >> >> >> >>

Speichern Sie dieses Programm unter dem Namen GDATA (DATEN erzeugen). Dann lassen Sie das Programm für die Werte, m = 5, a = 0, b = 100 laufen. Im RPN-Modus verwenden Sie:

5 SPC 0 SPC 1 0 0 GDATA

Die nachfolgende Abbildung stellt einen Boxplot der erzeugten Daten dar. Um den Graphen zu erhalten, kopieren Sie als erstes den erstellten Array, wandeln Sie diesen in einen Spaltenvektor, unter Verwendung von: OBJ→ 1 + →ARRY um (Die Funktionen OBJ→ und →ARRY finden Sie im Befehle Katalog, ↱ CAT). Speichern Sie mit Hilfe der Funktion STOΣ (ebenfalls über ↱ CAT zu finden) das Array in die Variable ΣDAT. Wählen

Sie Balken in TYPE für den Graphen, ändern Sie die Ansicht des Fensters auf H-VIEW (Ansicht): 0 32, V-VIEW (Ansicht): -10 10 und BarWidth (Balkenbreite) auf 1. Drücken Sie CANCL ON um zur Normalanzeige des Rechners zurückzukehren.

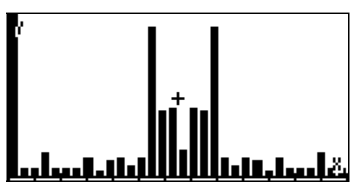


Um die FFT auf das Array in Stack Ebene 1 durchzuführen, verwenden Sie die Funktion FFT, welche im Menü MTH/FFT zu finden ist, auf das Array ΣDAT: ΣDAT FFT. Die Funktion FFT gibt ein Array von komplexen Zahlen, welche die Koeffizienten X_k der DFT darstellen, zurück. Die Magnitude der Koeffizienten X_k stellen ein Frequenzspektrum der ursprünglichen Daten dar. Um die Magnitude der Koeffizienten zu erhalten, wandeln Sie das Array in eine Liste um und wenden Sie anschließend die Funktion ABS auf die Liste an. Dies wird wie folgt erreicht: OBJ→ EVAL ◄ →LIST ↰ ABS

Letztendlich können Sie die Liste in eine Vektorspalte zurückkonvertieren und diese in ΣDAT wie folgt speichern:

OBJ→ 1 ENTER 2 →LIST →ARRY STOΣ

Um das Spektrum zu plotten, gehen Sie wie vorher bei Balkenplots beschrieben, vor. Der vertikale Bereich muss auf -1 bis 80 geändert werden. Das Spektrum für Frequenzen sieht wie folgt aus:



Das Spektrum zeigt zwei große Komponenten für zwei Frequenzen (diese sind die sinusförmigen Komponenten, sin (3x) und cos(5x)) und eine Anzahl kleinerer Komponenten für weitere Frequenzen.

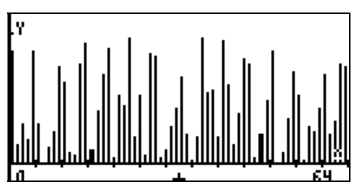
Beispiel 2 -- Um das Signal zu produzieren, welches das Spektrum gegeben wird, ändern wir das Programm GDATA um einen Absolutwert einzuschließen, damit es liest:

<< → m a b << '2^m' EVAL → n << '(b-a)/(n+1)' EVAL → Dx << 1 n FOR j 'a+(j-1)*Dx' EVAL f ABS NEXT n →ARRY >> >> >> >>

Speichern Sie diese Version des Programms unter GSPEC (SPECtrum erzeugen). Lassen Sie das Programm mit m = 6, a = 0, b = 100 Im RPN-Modus verwenden Sie:

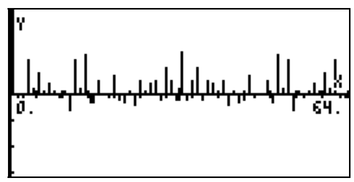
6 SPC 0 SPC 1 0 0 GSPEC

Wenn Sie fertig sind, drücken Sie die Taste ENTER, um zusätzlich eine Kopie des Array-Spektrums zu behalten. Konvertieren Sie diesen Zeilenvektor in einen Spaltenvektor und speichern Sie diesen in ∑DAT. Befolgen Sie die Vorgehensweise beim Erstellen eines Balkenplots, wird das erzeugte Spektrum für dieses Beispiel wie unten angezeigt aussehen. In diesem Fall liegt der horizontale Bereich zwischen 0 und 64, während der vertikale zwischen -1 und 10 liegt.

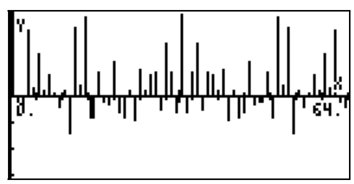


Benutzen Sie die Funktion IFFT, um das Signal, dessen Spektrum gezeigt wird, darzustellen. Da wir bereits eine Kopie des Spektrums im Stack (einen Zeilenvektor) belassen haben, müssen wir nur noch die Funktion IFFT im aus dem Menü MTH/FFT oder über den Befehle Katalog ↱ CAT aufrufen. Alternativ dazu können Sie einfach den Funktionsnamen eingeben, d.h. Sie tippen einfach ALPHA ALPHA I F F T ENTER. Das Signal wird als Array

(Zeilenvektor) mit komplexen Zahlen angezeigt. Uns interessiert aber lediglich der reelle Teil der Elemente. Benutzen Sie die Funktion RE aus dem Menü CMPLX (siehe Kapitel 4), um den reellen Teil der komplexen Zahlen zu extrahieren, z.B. tippen Sie ALPHA ALPHA R E ENTER. Das Resultat ist ein weiterer Zeilenvektor. Konvertieren Sie diesen in einen Spaltenvektor, speichern Sie diesen in SDAT und plotten diesen als Balkenplot, um das Signal anzuzeigen. Das Signal für dieses Beispiel wird unten, unter Verwendung eines horizontalen Bereiches von 0 bis 64 und einem Vertikalbereich von -1 bis 1, angezeigt.



Mit Ausnahme einer hohen Spitzen bei t = 0, ist das Signal vorwiegend Rauschen. Eine kleinere vertikale Skala (von -0,5 bis 0,5) zeigt das Signal wie folgt an:



Lösung für specifischen zweitrangigen Differential-gleichungen

In diesem Abschnitt stellen wir spezifische Sorten der normalen Differentialgleichungen dar und lösen die Differentialgleichungen deren Lösungen in einigen klassischen Funktionen definiert werden, z.B. Bessel's Funktionen, polynomische Hermite, usw. Die Beispiele werden im RPN-Modus dargestellt.

Die Cauchy oder Euler Gleichung

Eine Gleichung der Form $x^2 \cdot (d^2y/dx^2) + a \cdot x \cdot (dy/dx) + b \cdot y = 0$, wo a und b reale Konstanten sind, ist bekannt als die Cauchy oder Euler Gleichung. Eine Lösung für die Cauchy Gleichung können Sie finden wenn Sie annehmen daß $y(x) = x^n$.

Schreiben Sie die Gleichung so: 'x^2*d1d1y(x)+a*x*d1y(x)+b*y(x)=0' ENTER
Dann schreiben und ersetzen Sie die forgeschlagene Lösung: 'y(x) = x^n' ENTER
SUBST

Das Resultat ist: 'x^2*(n*(x^(n-1-1)*(n-1)))+a*x*(n*x^(n-1))+b*x^n =0, welches vereinfacht wird zu 'n*(n-1)*x^n+a*n*x^n+b*x^n = 0'. Das Teilen durch x^n, ergibt eine zusätzliche algebraische Gleichung:

'n*(n-1)+a*n+b = 0', oder

$$n^2 + (a-1) \cdot n + b = 0 .$$

- Wenn die Gleichung zwei unterschiedliche Wurzeln hat, sagen wir n_1 und n_2, dann ist die allgemeine Lösung dieser Gleichung $y(x) = K_1 \cdot x^{n_1} + K_2 \cdot x^{n_2}$.

- Wenn $b = (1-a)^2/4$, dann hat die Gleichung eine doppelte Wurzel $n_1 = n_2 = n = (1-a)/2$, und die Lösung zeigt sich $y(x) = (K_1 + K_2 \cdot \ln x)x^n$ zu sein.

Die Legendre Gleichung

Eine Gleichung der Form $(1-x^2)\cdot(d^2y/dx^2)-2\cdot x\cdot (dy/dx)+n\cdot (n+1) \cdot y = 0$, wo n eine richtige Nummer ist, ist bekannt als die Legendre Differentialgleichung. Jede mögliche Lösung für diese Gleichung ist bekannt als Legendre's Funktion. Wenn n eine nicht negative Ganzzahl ist, werden die Lösungen Legendre's Polyonomen genannt. Legendre's Polynom von Größe n wird gegeben bei

$$P_n(x) = \sum_{m=0}^{M} (-1)^m \cdot \frac{(2n-2m)!}{2^n \cdot m! \cdot (n-m)! \cdot (n-2m)!} \cdot x^{n-2m}$$

$$= \frac{(2n)!}{2^n \cdot (n!)^2} \cdot x^n - \frac{(2n-2)!}{2^n \cdot 1! \cdot (n-1)!(n-2)!} \cdot x^{n-2} + ... - ..$$

wo M = n/2 or (n-1)/2, welches eine Ganzzahl ist.

Legendre's Polynomen werden im Rechner vorprogrammiert und können durch das Verwenden der Funktion LEGENDRE zurückgerufen werden, die den Auftrag des Polynoms, n erteilt wird. Die Funktion LEGENDRE kann vom Befehl Katalog (↱ CAT) oder durch das Menü des Menüs ARITHMETIC/POLYNOMIAL erhalten werden (sehen Sie Kapitel 5). Die ersten sechs Legendre Polynomen werden wie folgt erreicht

0 Legendre, Resultat: 1, d.h., $P_0(x) = 1.0$.
1 Legendre, Resultat: 'X', d.h., $P_1(x) = x$.
2 Legendre, Resultat: '(3*X^2-1)/2', d.h., $P_2(x) = (3x^2-1)/2$.
3 Legendre, Resultat: '(5*X^3-3*X)/2', d.h., $P_3(x) =(5x^3-3x)/2$
4 Legendre, Resultat: '(35*X^4-30*X^2+3)/8', d.h., $P_4(x) =(35x^4-30x^2+3)/8$.
5 Legendre, Resultat: '(63*X^5-70*X^3+15*X)/8', d.h., $P_5(x) =(63x^5-70x^3+15x)/8$.

Das ODE $(1-x^2)\cdot(d^2y/dx^2)-2\cdot x\cdot (dy/dx)+[n\cdot (n+1)-m^2/(1-x^2)] \cdot y = 0$, hat für Lösung die Funktion $y(x) = P_n^m(x)= (1-x^2)^{m/2}\cdot(d^mPn/dx^m)$. Diese Funktion wird gekennzeichnet als eine dazugehörige Legendre Funktion.

Die Bessel Gleichung

Die gewöhnliche Differentialgleichung $x^2\cdot(d^2y/dx^2) + x\cdot (dy/dx)+ (x^2-\nu^2) \cdot y = 0$, wo der Parameter ν eine nichtnegative reale Zahl ist, ist bekannt als Bessel's Differentialgleichung. Lösungen zur Bessel's Gleichung werden in Bessel Funktionen der ersten Sorte der Folge ν ausgedruckt:

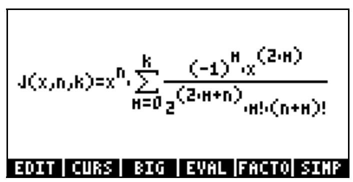
$$J_\nu(x) = x^\nu \cdot \sum_{m=0}^{\infty} \frac{(-1)^m \cdot x^{2m}}{2^{2m+\nu} \cdot m! \cdot \Gamma(\nu + m + 1)},$$

wo ν nicht eine Ganzzahl ist und die Funktion Gamma Γ(α) definiert in Kapitel 3.

Wenn ν = n, eine Ganzzahl, werden die Bessel Funktionen der ersten Sorte für n = Ganzzahl definiert durch:

$$J_n(x) = x^n \cdot \sum_{m=0}^{\infty} \frac{(-1)^m \cdot x^{2m}}{2^{2m+n} \cdot m! \cdot (n+m)!}.$$

Unabhängig davon, ob wir n (Nichtganzzahl) oder n (Ganzzahl) im Rechner verwenden, können wir die Bessel Funktionen der ersten Sorte definieren, indem wir die folgende begrenzte Reihe verwenden:



So haben wir Steuerung über dem Funktion's Auftrag, n und über dem Zahl der Elementen in der Reihe, k. Sobald Sie diese Funktion geschrieben haben, können Sie Funktion DEFINE verwenden, um Funktion J(x, n, k) zu definieren. Dieses verursacht das Variable [J] in den Weichmenü-schlüsseln. Z.B., um $J_3(0.1)$ mit 5 Bezeichnungen in der Reihe auszuwerten, errechnen Sie J(0.1,3,5). d.h. im RPN-Modus: [.] [1] [SPC] [3] [SPC] [5] [J] Das Resultat ist 2.08203157E-5

Wenn Sie einen Ausdruck für J_0 (x) erhalten möchten mit sagen wir 5 Bezeichnungen in der Reihe, brauchen Sie J(x, 0, 5). Das Resultat ist:

'1-0.25*x^3+0.015625*x^4-4.3403777E-4*x^6+6.782168E-6*x^8-6.78168*x^10'.

Für Nichtganzzahl Werte ν, wird diese Lösung zur Bessel Gleichung gegeben

$$y(x) = K_1 \cdot J_\nu(x) + K_2 \cdot J_{-\nu}(x).$$

Für Ganzzahl Werte sind die Funktionen Jn(x) und J-n(x) linear abhängig, seit

$$J_n(x) = (-1)^n \cdot J_{-n}(x),$$

folglich können wir sie nicht benutzen, um eine allgemeine Funktion zur Gleichung zu erhalten Stattdessen stellen wir die Bessel Funktionen der zweiten Sorte vor, die definiert wird wie

$$Y_\nu(x) = [J_\nu(x) \cos \nu\pi - J_{-\nu}(x)]/\sin \nu\pi,$$

für Nichtganzzahl ν, und für n Ganzzahl, mit n > 0 bei

$$Y_n(x) = \frac{2}{\pi} \cdot J_n(x) \cdot (\ln \frac{x}{2} + \gamma) + \frac{x^n}{\pi} \cdot \sum_{m=0}^{\infty} \frac{(-1)^{m-1} \cdot (h_m + h_{m+n})}{2^{2m+n} \cdot m! \cdot (m+n)!} \cdot x^{2m}$$

wo γ die Euler Konstante ist, definiert durch

$$-\frac{x^{-n}}{\pi} \cdot \sum_{m=0}^{n-1} \frac{(n-m-1)!}{2^{2m-n} \cdot m!} \cdot x^{2m}$$

und h_m die harmonische Reihe darstellt

$$\gamma = \lim_{r \to \infty} [1 + \frac{1}{2} + \frac{1}{3} + \ldots + \frac{1}{r} - \ln r] \approx 0.57721566490\ldots,$$

$$h_m = 1 + \frac{1}{2} + \frac{1}{3} + \ldots + \frac{1}{m}$$

Für der Fall n = 0, wird die Bessel Funktion der zweiten Sorte so definiert

$$Y_0(x) = \frac{2}{\pi} \cdot \left[J_0(x) \cdot (\ln \frac{x}{2} + \gamma) + \sum_{m=0}^{\infty} \frac{(-1)^{m-1} \cdot h_m}{2^{2m} \cdot (m!)^2} \cdot x^{2m} \right].$$

Mit diesen Definitionen wird eine allgemeine Lösung der Bessel Gleichung für alle Werte von ν vorbeigegeben

$$y(x) = K_1 \cdot J_\nu(x) + K_2 \cdot J_{-\nu}(x).$$

In einigen Fällen ist es notwendig, komplizierte Lösungen zu den Bessel Gleichungen zur Verfügung zu stellen, indem man die Bessel Funktionen der dritten Sorte der Folge definiert ν wie

$$H_n^{(1)}(x) = J_\nu(x) + i \cdot Y_\nu(x), \text{ and } H_n^{(2)}(x) = J_\nu(x) - i \cdot Y_\nu(x),$$

Diese Funktionen sind auch bekannt als die ersten und zweiten Hankel Funktionen der Folge ν.

In einigen Anwendungen, müssen Sie die sogenannten geänderten Bessel Funktionen der ersten Sorte der Folge auch verwenden können, definiert als $I_\nu(x) = i^{-\nu} \cdot J_\nu(i \cdot x)$, in dem i die Maßeinheit imaginäre Zahl ist. Diese Funktionen sind Lösungen zur Diffentialgleichung $x^2 \cdot (d^2y/dx^2) + x \cdot (dy/dx) - (x^2+\nu^2) \cdot y = 0$.

Die geänderten Bessel Funktionen der zweiten Sorte,

$$K_\nu(x) = (\pi/2) \cdot [I_{-\nu}(x) - I_\nu(x)]/\sin \nu\pi,$$

sind auch Lösungen für dieser ODE.

Sie können die Funktionen einführen, die Bessel Funktionen darstellen im Rechner in einer ähnlichen Weise zu der, die verwendet wird Bessel Funktionen der ersten Sorte zu definieren, aberdenken Sie daran, daß die endlosen Reihen im Rechner in eine begrenzte Reihe übersetzt werden müssen.

Chebyshev oder Tchebycheff Polynomen

Die Funktionen $T_n(x) = \cos(n \cdot \cos^{-1} x)$, und $U_n(x) = \sin[(n+1) \cos^{-1} x]/(1-x^2)^{1/2}$, n = 0, 1, … werden beziehungsweise Chebyshev oder Tchebycheff Polynomen der ersten und zweiten Sorte genannt. Die Polynomen Tn(x) sind Lösungen der Differentialgleichung $(1-x^2) \cdot (d^2y/dx^2) - x \cdot (dy/dx) + n^2 \cdot y = 0$.

Im Rechner erzeugt die Funktion TCHEBYCHEFF das Chebyshev oder Tchebycheff Polynom der ersten Sorte von Folge n, gegeben einen Wert von n > 0. Wenn die Ganzzahl n (n < 0) negativ ist, erzeugt die Funktion TCHEBYCHEFF ein Tchebycheff Polynom der zweiten Sorte der Folge n dessen Definition ist

$$U_n(x) = \sin(n\cdot\arccos(x))/\sin(\arccos(x)).$$

Sie können die Funktion TCHEBYCHEFF durch den Befehl Katalog (⮣ _CAT) zugänglich machen.

Die ersten vier Chebyshev oder Tchebycheff Polynomen der ersten und zweiten Sorte werden wie folgt erreicht:

0 TCHEBYCHEFF, Resultat: 1,	d.h., $T_0(x) = 1.0$.
-0 TCHEBYCHEFF, Resultat: 1,	d.h., $U_0(x) = 1.0$
1 TCHEBYCHEFF, Resultat: 'X',	d.h., $T_1(x) = x$.
-1 TCHEBYCHEFF, Resultat: 1,	d.h., $U_1(x) = 1.0$.
2 TCHEBYCHEFF, Resultat: '2*X^2-1,	d.h., $T_2(x) = 2x^2-1$.
-2 TCHEBYCHEFF, Resultat: '2*X',	d.h., $U_2(x) = 2x$
3 TCHEBYCHEFF, Resultat: '4*X^3-3*X',	d.h., $T_3(x) = 4x^3-3x$.
-3 TCHEBYCHEFF, Resultat: '4*X^2-1',	d.h., $U_3(x) = 24x^2-1$

Die Laguerre Gleichung

Die Laguerre Gleichung ist das zweitrangige, lineare ODE der Form form $x\cdot(d^2y/dx^2) +(1-x)\cdot (dy/dx) + n\cdot y = 0$. Laguerre Polynomen, definiert als

$$L_0(x) = 1, \quad L_n(x) = \frac{e^x}{n!}\cdot\frac{d^n(x^n\cdot e^{-x})}{dx^n}, n = 1,2,...$$

sind Lösungen zur Laguerre Gleichung. Laguerre Polynomen können auch berechnet werden mit:

$$L_n(x) = \sum_{m=0}^{n}\frac{(-1)^m}{m!}\cdot\binom{n}{m}\cdot x^m.$$

$$= 1 - n \cdot x + \frac{n(n-1)}{4} \cdot x^2 - ... + ... + \frac{(-1)^n}{n!} \cdot x^n$$

Die Bezeichnung

$$\binom{n}{m} = \frac{n!}{m!(n-m)!} = C(n,m)$$

ist das m-th Koeffizient der binomialen Expansion $(x+y)^n$. Sie stellt auch die Zahl Kombinationen der n Elemente dar, die hintereinander m genommen haben. Diese Funktion ist vorhanden im Rechner als Funktion COMB im MTH/PROB Menü (siehe auch Kapitel 17).

Sie können die folgende Funktion definieren, um Laguerre Polynomen zu errechnen:

$$L(x,n) = \sum_{m=0}^{n} \frac{(-1)^m}{m!} \cdot COMB(n,m) \cdot x^m$$

Wenn Sie es in der Gleichung Verfasser getippt haben, brauchen Sie die Funktion DEFINE, um die Funktion L(x, n) in Variable L zu verursachen.

Um den ersten vier Laguerre Polynomen zu erzeugen, benützen Sie L(x, 0), L(x, 1), L(x, 2), L(x, 3). Die Resultaten sind:

$L_0(x) = 1.0.$
$L_1(x) = 1\text{-}x.$
$L_2(x) = 1\text{-}2x+5x^2.$
$L_3(x) = 1\text{-}3x+1.5x^2\text{-}0.16666…x^3.$

Die Weber Gleichung und die Hermite Polynomen

Die Weber Gleichung wird wie folgt definiert $d^2y/dx^2+(n+1/2-x^2/4)y = 0$, for n = 0, 1, 2, … Eine bestimmte Lösung dieser Gleichung wird gegeben durch die Funktion $y(x) = \exp(-x^2/4)H^*(x/\sqrt{2})$, wo die Funktion $H^*(x)$ das Hermite Polynom ist:

$$H_0{}^* = 1, \quad H_n{}^*(x) = (-1)^n e^{x^2} \frac{d^n}{dx^n}(e^{-x^2}), \quad n = 1,2,..$$

Im Rechner die Funktion HERMITE, vorhanden durch das Menü ARITHMETIC/POLYNOMIAL. Funktion HERMITE nimmt als Argument eine Ganzzahl, n und bringt das Hermite Polynom des n-th Grads zurück. Z.B. werden die ersten vier Hermite Polynomen erreicht, indem man verwendet:

0 Hermite, Resultat: 1,	di.h., $H_0{}^* = 1$.
1 Hermite, Resultat: '2*X',	d.h., $H_1{}^* = 2x$.
2 Hermite, Resultat: '4*X^2-2',	d.h., $H_2{}^* = 4x^2-2$.
3 Hermite, Resultat: '8*X^3-12*X',	d.h., $H_3{}^* = 8x^3-12x$

Numerische und graphische Lösungen zu ODEs

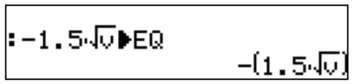
Differentialgleichungen, die nicht analytisch gelöst werden können, können numerisch oder graphisch gelöst werden, wie unten gezeigt wird.

Numerische Lösung zu erstrangige ODE

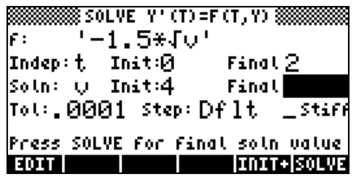
Durch den Gebrauch des numerischen Lösers (⮣ NUM.SLV), können Sie eine Eingang Form zugänglich machen, die Sie die erstrangigen, linearen gewöhnlichen Differentialgleichungen lösen läßt. Der Gebrauch von dieser Eigenschaft wird mit dem folgenden Beispiel dargestellt Die Methode, die in der Lösung verwendet wird, ist ein 4.-Auftrag Runge-Kutta Algorithmus.

Beispiel 1 -- Nehmen wir an, wir möchten die Differentialgleichung, $dv/dt = -1.5\ v^{1/2}$, mit $v = 4$ in $t = 0$ lösen. Wir werden gebeten, v für $t = 2$ zu finden.

Zuerst machen Sie den Ausdruck, der die Ableitung definiert und speichern Sie ihn in variables EQ. Die Abbildung nach links zeigt den ALG Modusbefehl, während die rechte seitliche Abbildung den RPN Stapel zeigt, bevor Sie STO▶ betätigen.



Dann tragen Sie das NUMERICAL SOLVER-Klima ein und wählen Sie den Differentialgleichunglöser vor. ↱ NUM.SLV ▼ OK Tragen Sie der folgenden Parameter ein:



Zum lösen, müssen Sie betätigen: SOLVE (bitte warten) EDIT. Das Resultat ist 0,2499 ≈ 0.25. Betätigen Sie OK.

Die Lösung wird presentiert als eine Tabelle von Werten.

Nehmen wir an, wir wollten eine Tabelle von Werten von v produzieren, für t = 0,00, 0,25, 2,00, gehen wir folgendermaßen vor:

Zuerst bereiten Sie eine Tabelle vor, um Ihre Resultate zu notieren. Notieren Sie in Ihrer Tabelle die Resultate Schritt für Schritt:

t	v
0.00	0.00
0.25	
…	…
2.00	

Zunächst, innerhalb des SOLVE, ändern Sie den abschließenden Wert der unabhängigen Variable bis 0,25, verwenden Sie:

▲ .25 OK ► ► SOLVE (bitte warten) EDIT

Lösungen für v an t = 0,25, v = 3,285.

OK INIT+ ▲ .5 OK ► ► SOLVE (bitte warten) EDIT

Ändern Sie Ausgangswert von t bis 0,25, und abschließender Wert von t bis 0,5, lösen Sie wieder für v(0,5) = 2,640.

OK EDIT ▲ .75 OK ▶ ▶ SOLVE (bitte warten) EDIT

Ändern Sie Ausgangswert von t bis 0,5, und abschließender Wert von t bis 0,75, lösen Sie wieder für v(0,75) = 2,066.

OK EDIT ▲ 1 OK ▶ ▶ SOLVE (bitte warten) EDIT

Ändern Sie Ausgangswert von t bis 0,75, und abschließender Wert von t bis 1, lösen Sie wieder für v(1) = 1,562.

Wiederholen Sie das Beispiel für t = 1,25, 1,50, 1,75, 2,00. Drücken Sie die Taste OK nachdem Sie das letzte Ergebnis in EDIT angezeigt haben. Um zur Normalanzeige zurückzukehren, drücken Sie ON oder NXT OK . Die verschiedenen Lösungen werden im Stack angezeigt, wobei das letzte Ergebnis in Ebene 1 angezeigt wird.

Das endgültige Ergebnis sieht wie folgt aus (aufgerundet in der dritten Dezimalstelle):

t	v
0.00	4.000
0.25	3.285
0.50	2.640
0.75	2.066
1.00	1.562
1.25	1.129
1.50	0.766
1.75	0.473
2.00	0.250

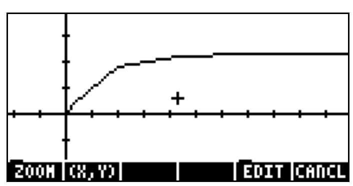
Graphische Lösung zu erstrangige ODE

Wenn wir nicht eine verschlossene Formlösung für das Integral erreichen können, können wir das Integral immer plotten, indem wir `Diff Eq` auf dem `TYPE`-Gebiet des PLOT-Klimas vorwählen. Daß geht wie folgt: Nehmen wir an, daß wir das Position x(t) für eine Geschwindigkeitsfunktion v(t) = $\exp(-t^2)$, plotten möchten mit x = 0 an t = 0. Wir wissen dort keine verschlossene

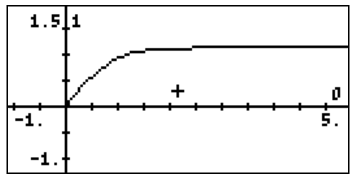
Formausdruck für das Integral, jedoch wissen wir, daß die Definition von v(t) dx/dt = exp(-t^2) ist.

Der Rechner läßt das Plotten der Lösung der Differentialgleichungen von der Form Y'(T) = F(T, Y) zu. Für unseren Fall ließen wir Y = x und T = t folglich F(T, y) = f(t, x) = exp(-t^2). Lassen wir die Lösung x(t) t = 0 bis 5 plotten, indem wir die folgende Tastenanschlagreihenfolge verwenden:

- ← 2D/3D (gleichzeitig, wenn im RPN Modus) um PLOT-Klima einzutragen.
- Heben Sie das Feld vor `TYPE` mit den Tasten ▲▼ hervorDann pressen Sie CHOOS, und müssen Sie `Diff Eq` hervorheben mit den ▲▼ Tasten Betätigen Sie OK.
- Tragen Sie die Funktion f(t, x) in die korrekte Position in der Eingangsform ein.
- Überprüfen Sie, ob die folgenden Parameter eingestellt werden auf: `H-VAR: 0, V-VAR: 1`
- Ändern Sie das unabhängige Variable zu t.
- Nehmen Sie Änderungen am PLOT-SETUP an: NXT OK.
- ← WIN (gleichzeitig, wenn im RPN Modus). um PLOT WINDOW-Klima einzutragen.
- Ändern Sie das horizontale und vertikale Ansichtfenster zu den folgenden Einstellungen: `H-VIEW: -1        5; V-VIEW: -1    1.5`
- Benutzen Sie auch folgende Werte für die noch verbleibenden Parameter: Init: 0, Final: 5, Step: Default, Tol: 0.0001, Init-Soln: 0
- Den Diagrammgebrauch plotten : ERASE DRAW



Wenn Sie das Diagramm beobachten, das geplottet wird, sehen Sie daß das Diagramm nicht sehr glatt ist. Das ist, weil der Plotter einen Zeitschritt verwendet, der zu groß ist. Um das Diagramm zu verfeinern und es glatter zu bilden, verwenden Sie einen Schritt von 0,1. Betätigen Sie CANCL und änderen Sie den Schrittwert: bis 0,1, verwenden Sie ERASE DRAW es dann nochmals, um das Diagramm zu wiederholen. Sie brauchen mehr Zeit den Plot durchzuführen, aber die Form ist sicherlich glatter als vorher. Versuchen Sie folgendes: EDIT NXT LABEL MENU damit Sie Axtkennzeichen und -strecke sehen.



Beachten Sie, daß die Markierungen für die Äxte als 0 (Horizontal) und 1 (Vertikal).gezeigt werden. Diese sind die Definitionen für die Äxte, wie in gegeben Schirm (← 2D/3D) d.h. H-VAR (t): 0, and V-VAR(x): 1.

NXT NXT PICT Menü zurückgewinnen und zum PICT Klima zurückgehen.
(X,Y) Koordinaten irgendeines Punktes auf dem Diagramm feststellen.

Verwenden Sie ◀ ▶ Tasten, zum verschieben des Cursors im Plotbereich. Am unteren Bildschirmrand sehen Sie die Koordinaten des Cursors wie (X,Y) d.h. der Rechner benützt X und Y als die Rückstellungsnamen für beziehungsweise die horizontalen und vertikalen Äxte. Betätigen Sie NXT CANCL um das Menü zurückzugewinnen und zum PLOT-WINDOW zurückzugehen. Schließlich pressen Sie ON, um zum Stapel zurückzugehen.

Numerische Lösung zu zweitrangige ODE

Integration von zweitrangigem ODEs kann vollendet werden, indem man die Lösung als Vektor definiert. Als Beispiel nehmen wir an, daß ein Frühling-Masse System abhängig von einer Dämpfungkraft ist, die zu seiner Geschwindigkeit proportional ist. Die resultierende Differentialgleichung ist:

$$\frac{d^2x}{dt^2} = -18.75 \cdot x - 1.962 \cdot \frac{dx}{dt}, \text{ oder,}$$

$$x'' = -18.75\ x - 1.962\ x',$$

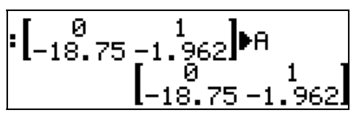
abhängig von den Ausgangsbedingungen v = x' = 6, x = 0, an t = 0. Wir möchten x, x' an t = 2 finden.

Schreiben Sie das ODE aufs neue: **w**' = **Aw**, wo **w** = [x x']^T, und **A** ist die Matrix 2 x 2, die oben gezeigt wird.

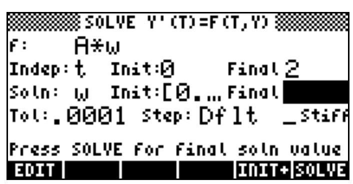
$$\begin{bmatrix} x \\ x' \end{bmatrix}' = \begin{bmatrix} 0 & 1 \\ -18.75 & -1.962 \end{bmatrix} \cdot \begin{bmatrix} x \\ x' \end{bmatrix}$$

Die Ausgangsbedingungen werden jetzt als **w** = [0   6]^T, for t = 0. Geschrieben. (Anmerkung: Das Symbol [ ]^T das umstellen des Vektors oder der Matrix).

Um dieses Problem zu lösen, machen Sie, und speichern Sie die Matrix **A**. z.B. im ALG-Modus:



Dann aktivieren Sie den numerischen Differentialgleichunglöser, in dem Sie verwenden: NUM.SLV ▼ OK. Um die Differentialgleichung mit Anlaßzeit t = 0 und abschließender Zeit t = 2 zu lösen, sollte die Eingang Form für den Differentialgleichunglöser wie folgt aussehen (beobachten Sie das Init: Wert für die Lösung: ist ein Vektor [0, 6]):



Betätigen Sie SOLVE (bitte warten) EDIT zum lösen für w(t=2). Die Lösung liest [.16716… -.6271…], i.e., x(2) = 0.16716, und x'(2) = v(2) = -0.6271. Betätigen Sie CANCL damit Sie zurückkehren zur SOLVE Klima.

Die Lösung wird presentiert als eine Tabelle von Werten.

Für das vorhergehende Beispiel waren wir nur interessiert die Werte zu finden der Position und der Geschwindigkeit t zu gegebener Zeit. Wenn wir eine Tabelle von Werten von x und von x', produzieren wollten t = 0,00, 0,25, 2,00, gehen wir folgendermaßen vor: Zuerst bereiten Sie eine Tabelle vor, um Ihre Resultate zu notieren.

t	x	x'
0.00	0.00	6.00
0.25		
…	…	…
2.00		

Zunächst, innerhalb des SOLVE KLimas, ändern Sie den abschließenden Wert der unabhängigen Variable bis 0,25, verwenden Sie:

▲ .25 OK ▶ ▶ SOLVE (bitte wait) EDIT

Lösungen für w an t = 0,25, w = [0,968 1,368].

OK INIT+ ▲ .5 OK ▶ ▶ SOLVE (bitte warten) EDIT

Ändern Sie Ausgangswert von t bis 0,25, und abschließender Wert von t bis 0,5, lösen Sie wieder für w(0,5) = [0,748 –2,616]

OK INIT+ ▲ .75 OK ▶ ▶ SOLVE (bitte warten) EDIT

Ändern Sie Ausgangswert von t bis 0,5, und abschließender Wert von t bis 0,75, lösen Sie wieder für w(0,75) = [0,0147 –2,859]

OK INIT+ ▲ 1 OK ▶ ▶ SOLVE (bitte warten) EDIT

Ändern Sie Ausgangswert von t bis 0,75, und abschließender Wert von t bis 1, lösen Sie wieder für w(1) =
[-0,469 -0,607]

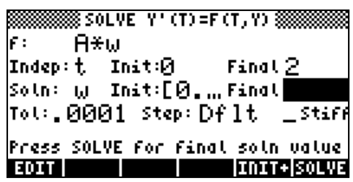
Wiederholen Sie das Beispiel für t = 1,25, 1,50, 1,75, 2,00. Drücken Sie die Taste OK nachdem Sie das letzte Ergebnis in EDIT angezeigt haben. Um zur Normalanzeige zurückzukehren, drücken Sie ON oder NXT OK. Die

verschiedenen Lösungen werden im Stack angezeigt, wobei das letzte Ergebnis in Ebene 1 angezeigt wird.
Das Endresultat sieht wie folgt aus:

t	x	x'		t	x	x'
0.00	0.000	6.000		1.25	-0.354	1.281
0.25	0.968	1.368		1.50	0.141	1.362
0.50	0.748	-2.616		1.75	0.227	0.268
0.75	-0.015	-2.859		2.00	0.167	-0.627
1.00	-0.469	-0.607				

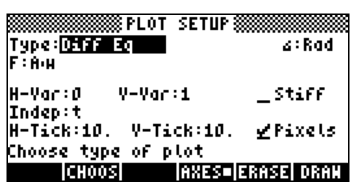
Graphische Lösung für ein zweitrangige ODE

Beginnen Sie damit, der numerischen Löser der Differentialgleichung zu aktivieren, ⊡ NUM.SLV ▼ OK. Der SOLVE Schirm sollte wie dieses aussehen;



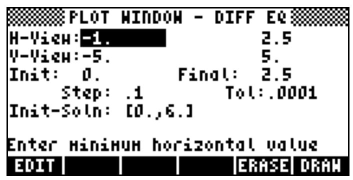
Beachten Sie, daß der Ausgangszustand für die Lösung (Soln: w Init:[0., …) umfaßt den Vektor [0, 6]. Betätigen Sie NXT OK.

Zunächst pressen Sie ⊡ 2D/3D (gleichzeitig, wenn im RPN Modus) um der PLOT-Klima einzutragen . Heben Sie das Feld vor `TYPE` mit den Tasten hervor ▲▼. Dann pressen Sie CHOOS, und müssen Sie `Diff Eq` hervorheben mit den ▲▼ Tasten . Betätigen Sie OK. Ändern Sie den Rest des PLOT-SETUP-Schirmes, um wie dieses auszusehen:



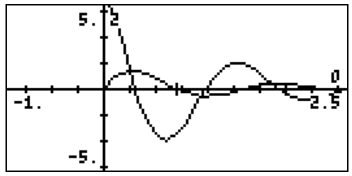
Beobachten Sie daß die Option V-Var: ist eingestellt auf 1, damit anzeichend, daß das erste Element in der Vektorlösung nämlich x' ist, gegen die unabhängige Variable t geplottet werden muss. Nehmen Sie Änderungen am PLOT SETUP an, weil Sie NXT OK betätigen .

Zunächst pressen Sie ← WIN (gleichzeitig, wenn im RPN Modus) um das PLOT WINDOW-Fenster zu sehen . Ändern Sie diese Eingang Form, um wie dieses auszusehen:



Das x' plotten gegen t Diagrammgebrauch: ERASE DRAW . Das x' gegen t plotten sieht so aus:

Um die zweite Kurve zu plotten, müssen wir die PLOT-SETUP Eingang Form noch einmal benutzen. Um diese Form vom Diagramm hier oben zu erreichen, benützen Sie: CANCL NXT OK ← 2D/3D (gleichzeitig, wenn im RPN Modus). Ändern Sie den Wert des V-Var: Das Feld auf 2 und DRAW betätigen (nicht ERASE oder Sie würden das Diagramm lösen, das oben produziert wurde). Verwenden Sie: EDIT NXT LABEL MENU damit Sie Axtmarken und -strecke sehen. Beachten Sie, daß der x Mittelliniemarke die Nr. 0 ist (die unabhängige Variable anzeigend), während der Y-axismarke die Nr. 2 ist (damit die zweite Variable anzeigend d.h. die letzte Variable geplottet). Das kombinierte Diagramm sieht wie dieses aus:



Betätigen Sie NXT NXT PICT CANCL ON um zur normalen Rechneranzeige zurückzugehen.

Numerische Lösung für unverwandte erstrangige ODE

Beobachten Sie der ODE: dy/dt = -100y+100t+101, abhängig von dem Ausgangszustand y(0) = 1.

Die genaue Lösung:

Diese Gleichung kann geschrieben werden als dy/dt + 100 y = 100 t + 101, und werden gelöst indem Sie einen integrierenden Faktor benützen wie folgendes:

‘(100*t+101)*EXP(100*t)’ ENTER ‘t’ ENTER RISCH

Das Resultat ist ‘(t+1)*EXP(100*t)’.

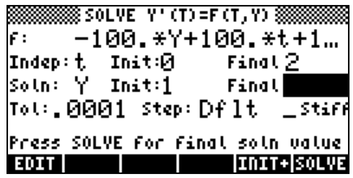
Nächstes fügen wir einen Integrationskonstante hinzu, dabei benütend: ‘C’ ENTER +

Dann teilen wir durch FI(x), indem wir verwenden: ‘EXP(100*t)’ ENTER ÷ .

Das Resultat ist: ‘((t+1)*EXP(100*t)+C)/EXP(100*t)’, i.e., $y(t) = 1+ t +C\cdot e^{100t}$. Gebrauch des Ausgangszustandes y(0) = 1, resultiert in $1 = 1 + 0 + C\cdot e^0$, oder C = 0, die bestimmte Lösung ist y(t) = 1+t .

Numerische Lösung

Wenn wir eine direkte numerische Lösung der ursprünglichen Gleichung dy/dt = -100y+100t+101 versuchen, der numerischen Löser des Rechners verwendend, werden wir sehen, daß der Löser, eine ungeregelte Zeitmenge scheint zu brauchen, wenn er die Gleichung löst. Um dieses zu überprüfen, stellen Sie Ihren numerischen Löser der Differentialgleichung ein auf solver (NUM.SLV OK)

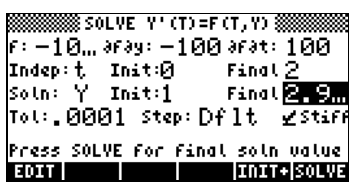


Hier versuchen wir, den Wert von y(2) y(0) = 1 gegeben zu erhalten. Mit der Lösung: Abschließendes Feld hervorgehoben, betätigen Sie SOLVE. Sie können prüfen, daß es keine Lösung gibt nach 2 Minuten. Betätigen Sie ON um die Berechnung zu annullieren.

Dieses ist ein Beispiel von eine unverwandten normalen Differetialgleichung. Ein unverwandte ODE ist eins dessen allgemeine Lösung Bestandteile enthält, die mit weit unterschiedlicher Rate unter die gleiche Stufensprung in der unabhängigen Variable varieren. In diesem bestimmten Fall enthält die allgemeine Lösung y(t) = 1+ t +C·e^{100t}, die Bestandteilen 't' und 'C·e^{100t}', welche varieren mit sehr unterschiedlicher Rate, außer den Fällen C=0 oder C≈0 (z.B., für C = 1, t = 0,1 C·e^{100t} =22026).

Der ODE des Rechners numerischen Löser läßt die Lösung von unverwandten ODEs zu, indem Sie die Wahl vorwählen `_Stiff` in dem Schirm SOLVE Y'(T) = F(T, Y). Wenn diese Wahl vorgewählt ist, müssen Sie die Werte von ∂f/∂y und ∂f/∂t zur Verfügung stellen. Für den Fall in Erwägung genommen ∂f/∂y =-100 und ∂f/∂t = 100.

Tragen Sie jene Werte auf den entsprechenden Gebieten ein vom SOLVE Schirm `Y'(T) = F(T,Y)`.



Wenn Sie fertig sind, bewegen Sie den Cursor zum abschließenden Feld und Sie betätigen SOLVE. Betätigen Sie EDIT um die Lösung zu sehen: 2.9999999999, d.h., 3.0.

Anmerkung: Die Option `Stiff` ist,auch für graphische Lösungen von Differentialgleichungen vorhanden.

Numerische Lösung für ODEs mit dem SOLVE/DIFF Menü.

Im RPN-Modus wird das Funktionsmenü SOLVE über 74 MENU gestartet. Dieses Menü wird im Detail in Kapitel 6 erläutert. Eines der Untermenüs, DIFF, enthält Funktionen für die numerische Lösung gewöhnlicher Differentialgleichungen zur Verwendung in der Programmierung. Diese Funktionen werden als nächstes im RPN-Modus mit System Flag 117 auf SOFT Menüs eingestellt, beschrieben.

Die Funktionen die zur Verfügung gestellt sind vom SOLVE/DIFF Menü sind die folgenden :

Die Funktion RKF

Diese Funktion wird benützt um die Lösung zu errechnen von einen Problem im Ausgangswert für eine erstrangige Differentialgleichung mit dem Runge-Kutta-Fehlbert 4^{th}. -5^{th}. Folge Lösung Entwurf. Nehmen Sie mal an, daß die Differentialgleichung die gelöst werden muss, durch dy/dx = f(x, y), mit y = 0 an x = 0 gegeben wird, und daß Sie Kriterien einer Konvergenz e für die Lösung erlauben. Sie können eine Stufensprung in der unabhängigen Variable Δx auch spezifizieren, durch die Funktion verwendet zu werden. Um diese Funktion laufen zu lassen bereiten Sie Ihren Stapel vor wie folgt:

3: {'x', 'y', 'f(x,y)'}
2: { ε Δx }
1: x_{final}

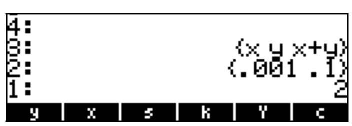
Der Wert im ersten Stapelniveau ist der Wert der unabhängigen Variable, in der Sie Ihre Lösung finden möchten, d.h. Sie möchten finden, $Y_{final} = f_s(X_{final})$, wo $f_s(x)$ die Lösung darstellt zur Diffentialgleichung. Der zweite Stapelniveau kann nur den Wert von ε, e enthalten, und der Schritt Δx wird als kleiner Default-Wert genommen. Nachdem Sie die Funktion RKF, laufen ließ, zeigt der Stapel die Linien:

2: {'x', 'y', 'f(x,y)'}
1: ε

Der Wert der Lösung, Y_{final}, ist vorhanden in variablen y. Diese Funktion ist für die Programmierung angebracht, da sie die Differentialgleichungspezifikationen und die Toleranz im Stapel läßt, der zu einer neuen Lösung bereit ist. Beachten Sie, daß die Lösung die Initiale Zustände x = 0 an y = 0 verwendet. Wenn Ihre tatsächlichen Ausgangslösungen $x = x_{init}$ an $y = y_{init}$ sind, können Sie diese Werte der Lösung immer hinzufügen, die von RKF zur Verfügung gestellt werden und daran denkend das folgende Verhältnis zu behalten:

RKF Lösung		Tatsächliche Lösung	
x	y	x	y
0	0	x_{init}	y_{init}
x_{final}	y_{final}	$x_{init} + x_{final}$	$y_{init} + y_{final}$

Die folgenden Schirmen zeigen den RPN Stapel vor und nach anwenden von Funktion RKF für die Differentialgleichung dy/dx = xy, ε = 0.001, Δx = 0.1.



Nach die Anwendung von Funktion RKF, enthält das Variable y den Wert 4.3880...

Die Funktion RRK

Diese Funktion ist der RKF Funktion ähnlich, außer daß RRK (Rosenbrock und Runge-Kutta Methoden) als Liste in Stapelniveau 3 für Eingang nicht nur die

Namen der unabhängigen und abhängigen Variablen und der Funktion, welche die Differentialgleichung definiert, erfordert, aber auch die Ausdrücke für die ersten und zweiten Ableitungen des Ausdruckes. Also, der Eingangsstapel für diese Funktion wird so aussehen:

3: {'x', 'y', 'f(x,y)' '∂f/∂x' '∂f/vy' }
2: { ε Δx }
1: x_{final}

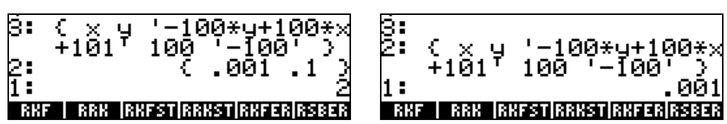
Der Wert im ersten Stapelniveau ist der Wert der unabhängigen Variable, in der Sie Ihre Lösung finden möchten, d.h. Sie möchten finden, $Y_{final} = f_s(X_{final})$, wo $f_s(x)$ die Lösung darstellt zur Diffentialgleichung. Der zweite Stapelniveau kann nur den Wert von ε, e enthalten, und der Schritt Δx wird als kleiner Default-Wert genommen. Nachdem Sie die Funktion RKF, laufen ließ, zeigt der Stapel die Linien:

2: {'x', 'y', 'f(x,y)' '∂f/∂x' '∂f/vy' }
1: { ε Δx }

Der Wert der Lösung, Y_{final}, ist vorhanden in variablen y.

Diese Funktion kann benützt werden um sogenannten unverwandtbare Differentialgleichungen zu lösen.

Die nachfolgende Anzeige zeigt den RPN Stack vor und nach anwenden der Funktion RRK:



Der in der Variable y gespeicherte Wert lautet 3,00000000004.

Die Funktion RKFSTEP

Diese Funktion benutzt eine Eingangsliste, die der der Funktion RKF, sowie die Toleranz für die Lösung ähnlich sind, und ein möglicher Schritt Δx (dx) und bringt die gleiche Eingangsliste zurück, gefolgt von der Toleranz und eine Schätzung des folgenden Schrittes in der unabhängigen Variable. Die

Funktion bringt die Eingangsliste zurück, die Toleranz, und der nächtsten Schritt in dem unabhängigen Variable daß diese Toleranz erfüllt. Also sieht der Eingangsstapel so aus:

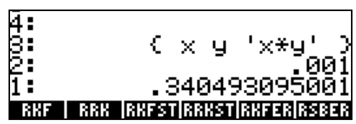
3: {'x', 'y', 'f(x,y)'}
2: ε
1: Δx

Nachdem Sie die Funktion laufen ließen, zeigt der Stapel die Linien:

3: {'x', 'y', 'f(x,y)'}
2: ε
1: $(\Delta x)_{next}$

Somit wird diese Funktion zur Ermittlung der angemessenen Größe eines Zeitschrittes, zur Erfüllung der gewünschten Toleranz, verwendet.

Die nachfolgende Anzeige zeigt den RPN Stack vor und nach anwenden der Funktion RKFSTEP:



Diese Ergebnisse zeigen, dass $(\Delta x)_{next}$ = 0.34049…

Die Funktion RRKSTEP

Die von dieser Funktion benutzte Eingabeliste ist der aus der Funktion RPK ähnlich, genauso die Toleranz für die Lösung, einen möglichen Schritt Δx und eine Zahl (LAST) in welcher die letzte in der Lösung gespeicherte Methode angegeben wird (1, wenn RFK benutzt wurde oder 2, wenn RRK benutzt wurde). Die Funktion RRKSTEP gibt dieselbe Eingabeliste, gefolgt von der Toleranz, einen Schätzwert für den nächsten Schritt in der unabhängigen Variablen und die aktuelle Methode (CURRENT) zum nächsten Schritt, zurück. Somit sieht der Eingabe Stack wie folgt aus:

4: {'x', 'y', 'f(x,y)'}
3: ε
2: Δx

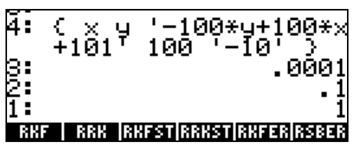
1: LAST

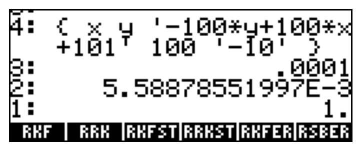
Nach Ausführen dieser Funktion sieht der Stack wie folgt aus:

4: {'x', 'y', 'f(x,y)'}
3: ε
2: $(\Delta x)_{next}$
1: CURRENT

Somit wird diese Funktion zur Ermittlung eines Zeitschrittes ($(\Delta x)_{next}$) entsprechend der benötigten Toleranz und die Methode, um zu diesem Ergebnis zu kommen (CURRENT) verwendet.

Die nachfolgende Anzeige zeigt den RPN Stack vor und nach anwenden der Funktion RRKSTEP:





Diese Ergebnisse zeigen dass $(\Delta x)_{next}$ = 0,00558… und die RKF Methode (CURRENT = 1) benutzt werden sollte.

Die Funktion RKFERR

Diese Funktion bringt die absolute Schätzung von Störungen für einen gegebenen Schritt zurück, wenn einen solchen Problem lösend, wie für Funktion RKF beschrieben wird . Also sieht der Eingangsstapel so aus:

2: {'x', 'y', 'f(x,y)'}
1: Δx

Nachdem Sie die Funktion laufen ließen, zeigt der Stapel die Linien:

4: {'x', 'y', 'f(x,y)'}
3: ε
2: Δy
1: Fehler

So wird diese Funktion benützt um die Stufensprung in der Lösung Δy festzustellen sowie den absoluten Fehler (Fehler).

In der nachfolgenden Abbildung ist der RPN Stack vor und nach anwenden der Funktion RKFERR zu sehen:

Diese Ergebnisse zeigen, dass Δy = 0,827… und error = -1,89…$\times 10^{-6}$.

Die Funktion RSBERR

Diese Funktion arbeitet ähnlich zu RKERR aber mit den Eingangselementen, die für die Funktion RRK verzeichnet werden. Also, der Eingangsstapel für diese Funktion wird so aussehen:

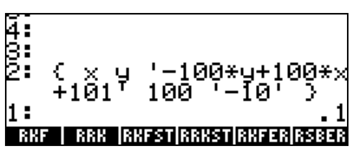
2: {'x', 'y', 'f(x,y)' '∂f/∂x' '∂f/vy' }
1: Δx

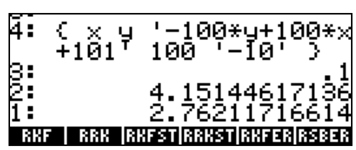
Nachdem Sie die Funktion laufen ließen, zeigt der Stapel die Linien:

4: {'x', 'y', 'f(x,y)' '∂f/∂x' '∂f/vy' }:
3: ε
2: Δy
1: Fehler

Der Wert der Lösung, Y_{final}, ist vorhanden in variablen .

Die nachfolgende Anzeige zeigt den RPN Stack vor und nach anwenden der Funktion RSBERR:





Diese Ergebnisse zeigen, dass Δy = 4,1514… und error = 2,762..., für Dx = 0,1 ist. Überprüfen Sie, ob Dx auf 0,01, Δy = -0,00307… und error = 0,000547 reduziert werden.

Kapitel 17
Wahrscheinlichkeitsanwendungen

In diesem Kapitel geben wir Beispiele für Anwendungen der Rechnerfunktionen zu Wahrscheinlichkeitsverteilungen.

Das MTH/PROBABILITY.. Untermenü - Teil 1

Das MTH/PROBABILITY.. Untermenü ist über die Tastenkombination ← MTH erreichbar. Wenn die Systemflag 117 auf die CHOOSE-Kästchen gesetzt ist, erscheint die folgende MTH-Optionsliste (siehe Abb. auf der linken Seite unten). Wir haben die Option PROBABILITY.. (Option 7) ausgewählt, um die folgenden Funktionen zu zeigen (siehe Abb. auf der rechten Seite unten):

In diesem Abschnitt besprechen wir die Funktionen COMB, PERM, ! (Fakultät), RAND und RDZ.

Fakultäten, Kombinationen und Permutationen

Die Fakultät eines Integers n wird definiert als: n! = n· (n-1) · (n-2)…3·2·1. Definitionsgemäß ist 0! = 1. Fakultäten werden bei der Berechnung von of Permutationszahlen und Objektkombinationen verwendet. Beispielsweise ist die Permutationszahl von r Objekten aus einer Menge mit n verschiedenen Objekten

$$_nP_r = n(n-1)(n-1)...(n-r+1) = n!/(n-r)$$

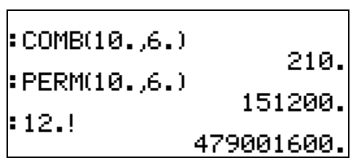
Außerdem beträgt die Anzahl der Kombinationen von n Objekten mit r Elementen auf einmal

$$\binom{n}{r} = \frac{n(n-1)(n-2)...(n-r+1)}{r!} = \frac{n!}{r!(n-r)!}$$

Um das System zu vereinfachen, verwenden Sie P(n,r) für Permutationen und C(n,r) für Kombinationen. Wir können Kombinationen, Permutationen, und Fakultäten mit den Funktionen COMB, PERM und ! aus dem Untermenü MTH/PROBABILITY.. berechnen. Die Bedienung dieser Funktionen wird im Folgenden beschrieben:

- COMB(n,r): Kombinationen von n Elementen mit r Elementen auf einmal
- PERM(n,r): Permutationen von n Elementen mit r Elementen auf einmal
- n!: Fakultät eines positiven Integers. Für einen Nicht-Integer gibt x! dann Γ(x+1) zurück, wobei Γ(x) die Gamma-Funktion ist (siehe Kapitel 3). Das Fakultätssymbol (!) kann auch als Tastenkombination ALPHA ↱ 2 eingegeben werden.

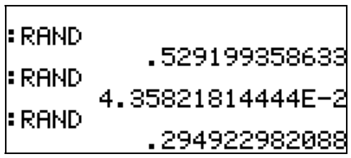
Ein Beispiel für Anwendungen dieser Funktionen wird im Folgenden gezeigt:

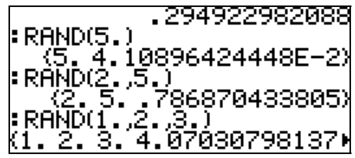


Zufallszahlen

Der Rechner bietet einen Zufallszahlengenerator, der einheitlich verteilte zufällige reele Zahlen zwischen 0 und 1 ausgibt. Der Generator kann zufällige Zahlensequenzen ausgeben. Nach einer bestimmten Anzahl von Malen jedoch (einer wirklich hohen Anzahl) tendiert die Sequenz dazu, sich selbst zu wiederholen. Aus diesem Grund ist der Zufallszahlengenerator eher ein Pseudozufallsgenerator. Um eine Zufallszahl mit Ihrem rechner zu erzeugen, verwenden Sie die Funktion RAND aus dem Untermenü MTH/PROBABILITY. Die folgende Anzeige zeigt eine Anzahl zufälliger Zahlen, die mit RAND erstellt wurden. Die Zahlen in der linken Abbildung sind mit der aufrufenden Funktion RAND ohne Argument erstellt worden.

Wenn Sie eine Argumentenliste in der Funktion RAND einsetzen, erhalten Sie die Zahlenliste plus eine zusätzliche Zufallszahl, die wie in der rechten Abbildung gezeigt angehängt wird.





Zufallszahlengeneratoren arbeiten im Allgemeinen so, dass sie einen Wert nehmen, der als “Ausgangszahl” des Generators bezeichnet wird, und einen mathematischen Algorithmus auf diesen "Ausgangszahl" ausführen, was eine neue (pseudo)zufällige Zahl erzeugt. Wenn Sie eine Zahlensequenz erzeugen wollen und diese später wiederholen möchten, können Sie den "Ausgangszahl" des Generators mit Verwendung der Funktion RDZ(n) ändern, wobei n der “Ausgangszahl” ist, bevor Sie die Sequenz erzeugen. Zufallszahlengeneratoren beginnen mit einer "Ausgangs"-Zahl, die in die erste Zufallszahl der Serie umgewandelt wird. Die aktuelle Zahl dient als "Ausgangszahl" für die nächste Zahl und so weiter. Durch erneutes Verwenden derselben "Ausgangszahl" für die Sequenz können Sie dieselbe Sequenz mehr als einmal erzeugen. Probieren Sie beispielsweise das Folgende aus:

RDZ(0.25) ENTER	Verwenden Sie 0.25 als "Ausgangszahl".
RAND() ENTER	Erste Zufallszahl = 0.75285…
RAND() ENTER	Zweite Zufallszahl = 0.51109…
RAND() ENTER	Dritte Zufallszahl = 0.085429….

Beginnen Sie die Sequenz erneut:

RDZ(0.25) ENTER	Verwenden Sie 0.25 als "Ausgangszahl".
RAND() ENTER	Erste Zufallszahl = 0.75285…
RAND() ENTER	Zweite Zufallszahl = 0.51109…
RAND() ENTER	Dritte Zufallszahl = 0.085429….

Um eine Sequenz zufälliger Zahlen zu erzeugen, verwenden Sie die Funktion SEQ. Um beispielsweise eine Liste mit 5 Zufallszahlen zu erzeugen, können

Sie im ALG-Modus: `SEQ(RAND(),1,5,1)` verwenden. Im RPN-Modus verwenden Sie nachfolgendes Programm:

« → n « 1 n FOR j RND NEXT n →LIST » »

Speichern Sie es in der Variablen RLST (Random LiST) und verwenden Sie VAR 5 RLST, um eine Liste mit 5 Zufallszahlen zu erzeugen.

Die Funktion RNDM(n,m) kann dazu verwendet werden, um eine Matrix mit n Reihen und m Spalten zu erzeugen, deren Elemente aus zufälligen Integeren zwischen -1 und 1 bestehen (siehe Kapitel 10).

Diskrete Wahrscheinlichkeitsverteilung

Eine zufällige Variable wird als diskret bezeichnet, wenn sie nur eine begrenzte Anzahl an Werten hat. So kann beispielsweise die Anzahl der Regentage an einem bestimmten Ort als diskrete Zufallsvariable betrachtet werden, weil wir diese nur als Integerzahlen zählen. Wenn X eine diskrete Zufallsvariable darstellt, wird ihre Wahrscheinlichkeitsverteilung (pmf) durch f(x) = P[X=x] dargestellt, d.h. die Wahrscheinlichkeit, dass die Zufallsvariable X den Wert x annimmt.

Die Wahrscheinlichkeitsverteilung muss die Bedingungen erfüllen, dass

f(x) >0, for all x,

und

$$\sum_{all\ x} f(x) = 1.0$$

Eine kumulative Verteilungsfunktion (cmd) wird definiert als

$$F(x) = P[X \le x] = \sum_{k \le x} f(k)$$

Als Nächstes definieren wir eine reihe von Funktionen, um diskrete Wahrscheinlichkeitsverteilungen zu berechnen. Wir schlagen vor, dass Sie ein

Unterverzeichnis anlegen, etwa HOME\STATS\DFUN (Diskrete FUNktionen), in dem wir die Wahrscheinlichkeitsverteilung und die Verteilungsfunktion für binomische und Poisson-Verteilungen berechnen.

Binomische Verteilung

Die Wahrscheinlichkeitsverteilung der binomische Verteilung ist gegeben durch

$$f(n,p,x) = \binom{n}{x} \cdot p^x \cdot (1-p)^{n-x}, \quad x = 0,1,2,\ldots,n$$

wobei $\binom{n}{x}$ = C(n,x) die Kombination von n Elementen mit x Elementen auf einmal ist. Die Werte n und p sind die Verteilungsparameter. Der Wert n stellt die Anzahl der Wiederholungen eines Experiments oder eine Beobachtung, die ein von zwei Ergebnissen haben können, z.B. Erfolg oder Misserfolg. Wenn die Zufallsvariable x die Anzahl der Erfolge in den n Wiederholunge darstellt, dann stellt p die Wahrscheinlichkeit, wie häufig ein Erfolg bei n gegebenen Wiederholungen auftreten kann, dar. Die Verteilungsfunktion für die binomische Verteilung ist gegeben durch

$$F(n,p,x) = \sum_{k=0}^{x} f(n,p,x), \quad x = 0,1,2,\ldots,n$$

Poisson-Verteilung

Die Wahrscheinlichkeitsverteilung der Poisson-Verteilung ist gegeben durch

$$f(\lambda,x) = \frac{e^{-\lambda} \cdot \lambda^x}{x!}, \quad x = 0,1,2,\ldots,\infty \,.$$

Wenn in diesem Ausdruck die Zufallsvariable X die Anzahl der Vorkommen eines Ereignisses oder einer Beobachtung pro Zeiteinheit, Länge, Fläche, Volumen usw. Darstellt, dann ist der von l dargestellte Parameter die durchschnittliche Anzahl von Vorkommen pro Zeiteinheit, Länge, Fläche, Volumen usw. Die Verteilungsfunktion für die Poisson-Verteilung ist gegeben durch

$$F(\lambda, x) = \sum_{k=0}^{x} f(\lambda, x), \quad x = 0,1,2,...,\infty$$

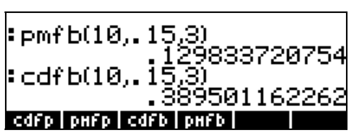
Verwenden Sie als nächstes die Funktion DEFINE (← DEF), um die folgenden Wahrscheinlichkeitsverteilungen (pmf) und Verteilungsfunktionen (cdf) zu definieren:

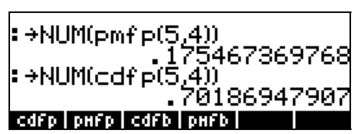
```
DEFINE(pmfb(n,p,x) = COMB(n,x)*p^x*(1-p)^(n-x))
DEFINE(cdfb(n,p,x) = Σ(k=1,x,pmfb(k,p,x)))
DEFINE(pmfp(λ,x) = EXP(-λ)*λ^x/x!)
DEFINE(cdfp(λ,x) = Σ(k=1,x,pmfb(λ,x)))
```

Die Funktionsnamen stehen für:

- pmfb: Wahrscheinlichkeitsverteilung für die binomische Verteilung
- cdfb: Verteilungsfunktion für die binomische Verteilung.
- pmfp: Wahrscheinlichkeitsverteilung für die Poisson-Verteilung
- cdfp: Verteilungsfunktion für die Poisson- Verteilung.

Beispiele für die Berechnungen, die diese Funktionen verwenden, werden im Folgenden gezeigt:





Stetige Wahrscheinlichkeitsverteilungen

Die Wahrscheinlichkeitsverteilung für eine stetige Zufallsvariable X wird durch eine Funktion f(x), die als Wahrscheinlichkeitsdichte (pdf) bekannt ist, charakterisiert. Die pdf hat die folgenden Eigenschaften: $f(x) > 0$, für alle x und

$$P[X < x] = F(x) = \int_{-\infty}^{x} f(\xi)d\xi.$$

$$\int_{-\infty}^{+\infty} f(x)dx = 1.$$

Wahrscheinlichkeiten werden mit der kumulativen Verteilungsfunktion (cdf), F(x), berechnet und definiert durch $P[X < x] = F(x) = \int_{-\infty}^{x} f(\xi)d\xi$,

wobei P[X<x] für "die Wahrscheinlichkeit, dass die Zufallsvariable X kleiner als der Wert x ist" steht.

In diesem Abschnitt beschreiben wir mehrere stetige Wahrscheinlichkeitsverteilungen, einschließlich der Gamma-, Exponential-, Beta- und Weibull-Verteilungen. Diese Verteilungen werden in jedem Statistikbuch beschrieben. Einige dieser Verteilungen verwenden die vorhin definierte Gammafunktion, die im Rechner mit der Fakultätsfunktion als $\Gamma(x) = (x-1)!$ für jede reelle Zahl x berechnet wird.

Die Gammaverteilung

Die Wahrscheinlichkeitsverteilung (pdf) für die Gammaverteilung ist gegeben durch

$$f(x) = \frac{1}{\beta^{\alpha}\Gamma(\alpha)} \cdot x^{\alpha-1} \cdot \exp(-\frac{x}{\beta}), for \quad x > 0, \alpha > 0, \beta > 0;$$

Die entsprechende (kumulative) Verteilung (cdf) würde durch ein Integral gegeben werden, dass keine Auflösung in geschlossener Form hat.

Die Exponentialverteilung

Die Exponentialverteilung ist die Gammaverteilung mit a = 1. Ihre pdf wird gegeben durch

$$f(x) = \frac{1}{\beta} \cdot \exp(-\frac{x}{\beta}), for \quad x > 0, \beta > 0,$$

während ihre cdf durch F(x) = 1 - exp(-x/β), for x>0, β >0 gegeben ist.

Die Betaverteilung

Die pdf für die Gammaverteilung ist gegeben durch

$$f(x) = \frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\cdot\Gamma(\beta)}\cdot x^{\alpha-1}\cdot(1-x)^{\beta-1}, for \quad 0 < x < 1, \alpha > 0, \beta > 0$$

Wie im Fall der Gammaverteilung ist die entsprechende cdf für die Betaverteilung auch durch ein Integral mit einer Auflösung ohne geschlossene Form gegeben.

Die Weibull-Verteilung

Die pdf für die Weibull-Verteilung ist gegeben durch

$$f(x) = \alpha\cdot\beta\cdot x^{\beta-1}\cdot\exp(-\alpha\cdot x^{\beta}), \quad for\ x > 0, \alpha > 0, \beta > 0$$

Während die entsprechende cdf gegeben ist durch

$$F(x) = 1 - \exp(-\alpha\cdot x^{\beta}), \quad for\ x > 0, \alpha > 0, \beta > 0$$

Funktionen für stetige Verteilungen

Um eine Funktionssammlung zu definieren, die den Gamma-, Exponential-, Beta- und Weibull-Verteilungen entsprechen, erstellen Sie zuerst ein Unterverzeichnis namens CFUN (Stetige FUNktionen) und definieren die folgenden Funktionen (wechseln Sie in den Approx-Modus):

Gamma-pdf:	`'gpdf(x) = x^(α-1)*EXP(-x/β)/(β^α*GAMMA(α))'`
Gamma-cdf:	`'gcdf(x) = ∫(0,x,gapd(t),t)'`
Beta-pdf:	`'βpdf(x)= GAMMA(α+β)*x^(α-1)*(1-x)^ (β-1)/(GAMMA(α)*GAMMA(β))'`
Beta-cdf:	`'βcdf(x) = ∫(0,x, βpd(t),t)'`
Exponential-pdf:	`'epdf(x) = EXP(-x/β)/β'`
Exponential-cdf:	`'ecdf(x) = 1 - EXP(-x/β)'`

Weibull-pdf: `'Wpdf(x) = α*β*x^(β-1)*EXP(-α*x^β)'`
Weibull-cdf: `'Wcdf(x) = 1 - EXP(-α*x^β)'`

Verwenden Sie die Funktion DEFINE, um diese Funktionen zu definieren. Geben Sie als nächstes die Wert für α und β ein, z.B.,

[1] [STO▶] [ALPHA] [→] [A] [ENTER] [2] [STO▶] [ALPHA] [→] [B] [ENTER]

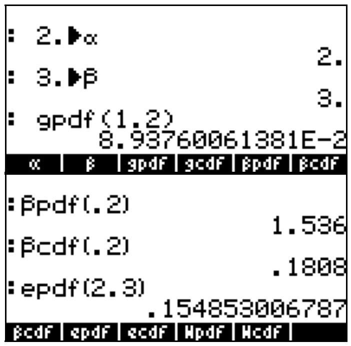
Zuletzt müssen Sie für die cdf für Gamma- und Beta-cdfs die Programmdefinitionen bearbeiten, um →NUM zu den durch die Funktion DEFINE erstellten Programme hinzuzufügen. So sollte beispielsweise die Gamma-cdf, d.h., die Funktion gcdf, verändert werden, um so angezeigt zu werden: `<< → x '∫ (0,x,gapd(t),t)' →NUM >>` und wieder in gespeichert werden. Wiederholen Sie diesen Vorgang für βcdf.

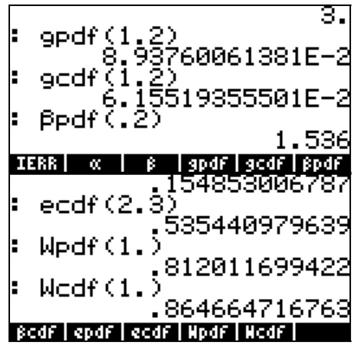
Anders als die vorhin definierten diskreten Funktionen enthalten die in diesem Abschnitt definierten stetigen Funktionen nicht ihre Parameter (α und/oder β) in ihren Definitionen. Deshalb müssen Sie diese nicht im Display eingeben, um die Funktionen zu berechnen. Diese Parameter müssen jedoch vorher definiert werden, indem die entsprechenden Wertein den Variablen α und β gespeichert werden. Wenn alle Funktionen und die Werte α und β gespeichert worden sind, können Sie die Menümarkierungen mit der Funktion ORDER zuordnen. Die Funktion wird nun wie folgt aufgerufen:

ORDER({'α','β','gpdf','gcdf','βpdf','βcdf','epdf','ecdf','Wpdf','Wcdf'})

Wenn Sie diesem Befehl folgen, zeigen die Menümarkierungen das Folgende (Drücken Sie [NXT], um zur zweiten [NXT]iste zu gelangen. Drücken Sie noch einmal [NXT], um wieder zur ersten [NXT]iste zu gelangen):

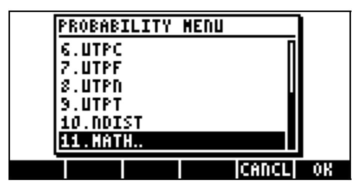
Einige Anwendungsbeispiele dieser Funktionen, für Werte von α = 2, β = 3, werden unten gezeigt. Beachten Sie dabei die Variable IERR, welche in der zweiten Abbildung erscheint. Diese stammt aus einer numerischen Integration der Funktion gcdf.





Stetige Verteilungen für statistische Inferenz

In diesem Abschnitt besprechen wir vier stetige Wahrscheinlichkeitsverteilungen, die allgemein für Probleme in Zusammenhang mit statistischer Inferenz verwendet werden. Diese Verteilungen sind die normale Verteilung, die studentische t-Verteilung, die Chi-Quadrat-Verteilung (χ^2) und die F-Verteilung. Die vom Rechner angebotenen Funktionen für diese Verteilungen, um Wahrscheinlichkeiten für diese Verteilungen zu bewerten, sind im Menü MTH/PROBABILITY weiter oben in diesem Kapitel enthalten. Die Funktionen sind NDIST, UTPN, UTPT, UTPC und UTPF. Ihre Anwendung wird in den folgenden Abschnitten beschrieben. Um diese Funktionen zu sehen, aktivieren Sie das Menü MTH: ← MTH und wählen die Option PROBABILITY:



Normale Verteilung pdf

Der Ausdruck für die normale Verteilung pdf lautet:

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} \exp[-\frac{(x-\mu)^2}{2\sigma^2}],$$

wo μ das Mittel ist und σ^2 die Verteilungsvarianz. Um den Wert von $f(\mu,\sigma^2,x)$ für die normale Verteilung zu berechnen, verwenden Sie die Funktion NDIST mit den folgenden Argumenten: das Mittel μ, die Varianz σ^2 und den Wert x ,d.h., NDIST(μ,σ^2,x). Überprüfen Sie das beispielsweise für eine normale Verteilung, f(1.0,0.5,2.0) = 0.20755374.

Normale Verteilung cdf

Der Rechner hat eine Funktion UTPN, die das obere Ende der normalen Verteilung berechnet, d.h., UTPN(x) = P(X>x) = 1 - P(X<x). Um den Wert des oberen Endes der normalen Verteilung UTPN zu erhalten, müssen wir die folgenden Werte eingeben: das Mittel μ, die Varianz σ^2 und den Wert x, z.B., UTPN((μ,σ^2,x)

Überprüfen Sie das beispielsweise für eine normale Verteilung mit μ = 1.0, σ^2 = 0.5, UTPN(0.75) = 0.638163. Verwenden Sie UTPN(1.0,0.5,0.75) = 0.638163.

Verschiedene Wahrscheinlichkeitsberechnungen für normale Verteilungen [X is N(μ,σ^2)] können unter Verwendung der Funktion UTPN wie folgt definiert werden:

- P(X<a) = 1 - UTPN(μ, σ^2,a)
- P(a<X<b) = P(X<b) - P(X<a) = 1 - UTPN(μ, σ^2,b) - (1 - UTPN(μ, σ^2,a)) = UTPN(μ, σ^2,a) - UTPN(μ, σ^2,b)
- P(X>c) = UTPN(μ, σ^2,c)

Beispiele: Bei Verwendung von μ = 1.5, and σ^2 = 0.5 ergibt sich:

P(X<1.0) = 1 - P(X>1.0) = 1 - UTPN(1.5, 0.5, 1.0) = 0.239750.

P(X>2.0) = UTPN(1.5, 0.5, 2.0) = 0.239750.

P(1.0<X<2.0) = F(1.0) - F(2.0) = UTPN(1.5,0.5,1.0) - UTPN(1.5,0.5,2.0) = 0.7602499 - 0.2397500 = 0.524998.

Die studentische t-Verteilung

Die studentische t- oder einfach die t-Verteilung hat einen Parameter ν, der als der Freiheitsgrad der Verteilung bekannt ist. Die Wahrscheinlichkeitsverteilung (pdf) ist gegeben durch

$$f(t) = \frac{\Gamma(\frac{\nu+1}{2})}{\Gamma(\frac{\nu}{2}) \cdot \sqrt{\pi\nu}} \cdot (1+\frac{t^2}{\nu})^{-\frac{\nu+1}{2}}, -\infty < t < \infty$$

wobei $\Gamma(\alpha) = (\alpha\text{-}1)!$ Die im Kapitel 3 definierte GAMMA-Funktion ist.

Der Rechner sieht Werte für das obere Ende der (kumulativen) Verteilung, für die t-Verteilung, die Funktion UTPT, wenn der Parameter ν und der Wert von t gegeben sind, d.h., UTPT(ν,t), vor. Die Definition dieser Funktion ist deshalb

$$UTPT(\nu,t) = \int_t^{\infty} f(t)dt = 1 - \int_{-\infty}^{t} f(t)dt = 1 - P(T \le t)$$

Zum Beispiel ist UTPT(5,2.5) = 2.7245…E-2. Andere Wahrscheinlichkeitsrechnungen für die t-Verteilung mit der Funktion UTPT wie folgt definiert werden:

- P(T<a) = 1 - UTPT(ν,a)
- P(a<T<b) = P(T<b) - P(T<a) = 1 - UTPT(ν,b) - (1 - UTPT(ν,a))
  = UTPT(ν,a) - UTPT(ν,b)
- P(T>c) = UTPT(ν,c)

Beispiele: Gegeben ist ν = 12, bestimme:

P(T<0.5) = 1-UTPT(12,0.5) = 0.68694..
P(-0.5<T<0.5) = UTPT(12,-0.5)-UTPT(12,0.5) = 0.3738…
P(T> -1.2) = UTPT(12,-1.2) = 0.8733…

Die Chi-Quadrat-Verteilung

Die Chi-Quadrat-Verteilung (χ^2) hat einen Parameter ν, der als Freheitsgrade bekannt ist. Die Wahrscheinlichkeitsverteilung (pdf) ist gegeben durch

$$f(x) = \frac{1}{2^{\frac{\nu}{2}} \cdot \Gamma(\frac{\nu}{2})} \cdot x^{\frac{\nu}{2}-1} \cdot e^{-\frac{x}{2}}, \nu > 0, x > 0$$

Der Rechner sieht Werte für das obere Ende der (kumulativen) Verteilung für die χ^2-Verteilung unter Verwendung von [UTPC] vor, wenn der Wert x und der Parameter ν gegeben sind. Die Definition dieser Funktion ist deshalb

$$UTPC(\nu, x) = \int_t^{\infty} f(x)dx = 1 - \int_{-\infty}^{t} f(x)dx = 1 - P(X \le x)$$

Zur Verwendung dieser Funktion benötigen wir die Freiheitsgrade ν und den Wert der Chi-Quadrat-Variable x, d.h., UTPC(ν,x). Zum Beispiel ist UTPC(5, 2.5) = 0.776495…

Verschiedene Wahrscheinlichkeitsberechnungen können für die Chi-Quadrat-Verteilung mit der Funktion UTPC wie folgt definiert werden:

- P(X<a) = 1 - UTPC(ν,a)
- P(a<X<b) = P(X<b) - P(X<a) = 1 - UTPC(ν,b) - (1 - UTPC(ν,a)) = UTPC(ν,a) - UTPC(ν,b)
- P(X>c) = UTPC(ν,c)

Beispiele: Gegeben ist ν = 6, bestimme:
P(X<5.32) = 1-UTPC(6,5.32) = 0.4965..
P(1.2<X<10.5) = UTPC(6,1.2)-UTPC(6,10.5) = 0.8717…
P(X> 20) = UTPC(6,20) = 2.769..E-3

Die F-Verteilung

Die F-Verteilung hat zwei Parameter νN = Zähler der Freiheitsgrade und νD = Nenner der Freiheitsgrade.Die Wahrscheinlichkeitsverteilung (pdf) ist gegeben durch

$$f(x) = \frac{\Gamma(\frac{\nu N + \nu D}{2}) \cdot (\frac{\nu N}{\nu D})^{\frac{\nu N}{2}} \cdot F^{\frac{\nu N}{2}-1}}{\Gamma(\frac{\nu N}{2}) \cdot \Gamma(\frac{\nu D}{2}) \cdot (1 - \frac{\nu N \cdot F}{\nu D})^{(\frac{\nu N + \nu D}{2})}}$$

Der Rechner sieht Werte für das obere ende der (kumulativen) Verteilung für die F-Verteilung, die Funktion UTPF, wenn die Parameter νN und νD gegeben sind, und den Wert von F vor. Die Definition dieser Funktion ist deshalb

$$UTPF(\nu N, \nu D, F) = \int_t^{\infty} f(F)dF = 1 - \int_{-\infty}^{t} f(F)dF = 1 - P(\Im \le F)$$

Zum Beispiel, um UTPF(10,5, 2.5) = 0.161834… zu berechnen

Verschiedene Wahrscheinlichkeitsberechnungen können für die F-Verteilung mit der Funktion UTPF wie folgt definiert werden:

- P(F<a) = 1 - UTPF(νN, νD,a)
- P(a<F<b) = P(F<b) - P(F<a) = 1 -UTPF(νN, νD,b)- (1 - UTPF(νN, νD,a))
  = UTPF(νN, νD,a) - UTPF(νN, νD,b)
- P(F>c) = UTPF(νN, νD,a)

Beispiel: Gegeben ist νN = 10, νD = 5, finde:

P(F<2) = 1-UTPF(10,5,2) = 0.7700…
P(5<F<10) = UTPF(10,5,5) – UTPF(10,5,10) = 3.4693..E-2
P(F>5) = UTPF(10,5,5) = 4.4808..E-2

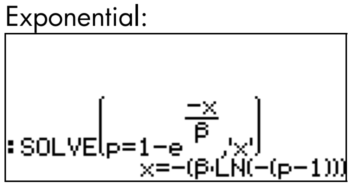
Inverse Verteilungsfunktionen

Für eine stetige Zufallsvariable X mit der kumulativen Dichtefunktion (cdf) F(x) = P(X<x) = p, müssen wir, um die inverse Verteilungsfunktion zu berechnen, den wert von x finden, so dass x = $F^{-1}(p)$. Dieser Wert ist relativ einfach zu finden in den Fällen der Exponential- und der Weibull-Verteilungen, da ihre cdfs einen Ausdruck mit geschlossener Form haben:

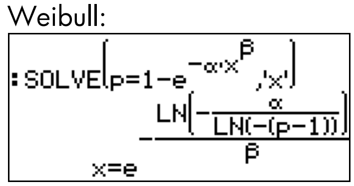
- Exponential, $F(x) = 1 - \exp(-x/\beta)$
- Weibull, $F(x) = 1-\exp(-\alpha x^{\beta})$

Um die inversen cdfs für diese beiden Verteilungen zu finden, müssen wir nur x aus diesen beiden Ausdrücken auflösen, d.h.,

Exponential:



Weibull:



Für die Gamma- und Beta-Verteilungen sind die aufzulösenden Ausdrücke komplizierter aufgrund der vorhandenen Integrale, d.h.

- Gamma, $p = \int_0^x \frac{1}{\beta^{\alpha}\Gamma(\alpha)} \cdot z^{\alpha-1} \cdot \exp(-\frac{z}{\beta})dz$
-
- Beta, $p = \int_0^x \frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\cdot\Gamma(\beta)} \cdot z^{\alpha-1} \cdot (1-z)^{\beta-1}\,dz$

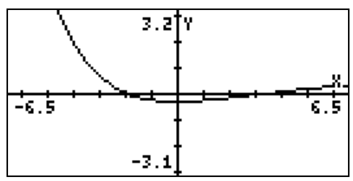
Eine numerische Auflösung mit dem numerischen Auflöser ist nicht machbar wegen des im Ausdruck enthaltenen Integralzeichens. Es ist jedoch eine graphische Lösung möglich. Details, wie Sie die Wurzel eines Graphen finden, werden in Kapitel 12 vorgestellt. Um numerische Ergebnisse zu gewährleisten, ändern Sie die Einstellung CAS auf Approx. Die zu zeichnende Funktion für die Gamma-Verteilung ist

$$Y(X) = \int(0,X,z^{\wedge}(\alpha-1)*\exp(-z/\beta)/(\beta^{\wedge}\alpha*\mathrm{GAMMA}(\alpha)),z)-p$$

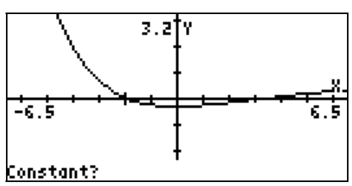
Für die Beta-Verteilung ist zu zeichnende Funktion

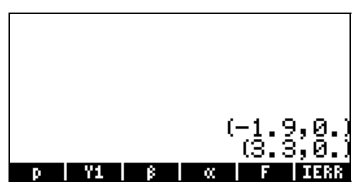
$$Y(X) = \int(0,X,z^{\wedge}(\alpha-1)*(1-z)^{\wedge}(\beta-1)*\mathrm{GAMMA}(\alpha+\beta)/(\mathrm{GAMMA}(\alpha)*\mathrm{GAMMA}(\beta)),z)-p$$

Um den Plot zu erstellen, ist es notwendig, die Werte von α, β und p zu speichern, bevor Sie einen Plot starten. Zum Beispiel ist für $\alpha = 2$, $\beta = 3$ und p = 0.3 der Plot von Y(X) für die Gamma-Verteilung der folgende. (Beachten Sie bitte, dass wegen der komplizierten Natur der Funktion Y(X), einige Zeit benötigt wird, bevor der Graph erzeugt wird. Bleiben Sie geduldig.)

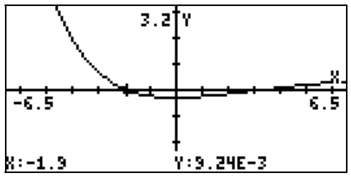


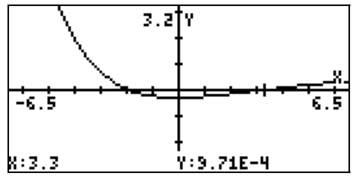
Es gibt zwei Wurzeln für diese Funktionbei Verwendung der Funktion ROOT innerhalb der Plotumgebung. Wegen des Integrals in der Gleichung wird die Wurzel angenähert und nicht in der Plotanzeige gezeigt. Sie erhalten auf der Anzeige nur die Nachricht Constant?. Wenn Sie aber an diesem Punkt ENTER drücken, wird die Annäherungswurzel in der Anzeige aufgeführt. Es werden in der rechten Abbildung unten zwei Wurzeln gezeigt.





Alternativ können Sie die Funktion TRACE (X,Y) verwenden, um die Wurzeln durch Verfolgen der Kurve nahe ihrer Schnittpunkte mit der X-Achse zu berechnen. Zwei Berechnungen werden unten gezeigt:





Diese Berechnungen schlagen die Lösungen x = -1.9 und x = 3.3 vor. Sie können diese "Lösungen" durch Auswertung der Funktion Y1(X) für X = -1.9 und X = 3.3 überprüfen, d.h.,

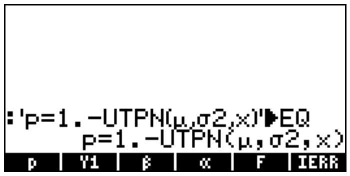
```
:Y1(-1.9)
             .009244101213
:Y1(3.3)
             .000970724234
  p  |  Y1  |  β  |  α  |  F  | IERR
```

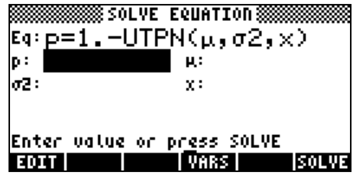
Für die Normal-, studentischen t-, Chi-Quadrat- (χ^2) und F-Verteilungen, die durch die Funktionen UTPN, UTPT, UPTC und UTPF im rechner dargestellt werden, finden Sie die inversen Antwort durch Lösung einer der folgenden Gleichungen finden:

- Normal, p = 1 – UTPN(μ,σ2,x)
- studentische t, p = 1 – UTPT(ν,t)
- Chi-Quadrat, p = 1 – UTPC(ν,x)
- F-Verteilung: p = 1 – UTPF(νN,νD,F)

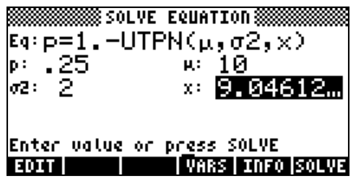
Wir weisen darauf hin, dass der zweite Parameter in der UTPN-Funktion σ2 ist, nicht σ^2, und die Verteilungsvarianz darstellt. Auch ist das Symbol ν (der kleingeschriebene griechische Buchstabe no) im Rechner nicht verfügbar. Sie können beispielsweise das γ (Gamma) statt des ν verwenden. Der Buchstabe γ ist über den Zeichensatz (↱ CHARS) verfügbar.

Um beispielsweise den Wert von x für eine Normalverteilung mit μ = 10, σ^2 = 2, mit p = 0.25, zu erhalten, speichern Sie die Gleichung 'p=1-UTPN(μ,σ2,x)' in der Variablen EQ (Abbildung auf der linken Seite unten). Dann starten Sie den numerischen Löser, um das Eingabefeld in der Abbildung auf der Seite zu erhalten:



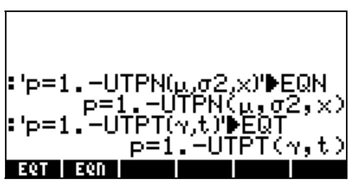


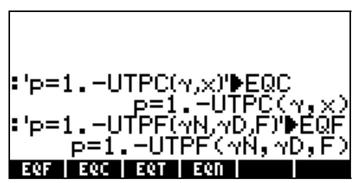
Der nächste Schritt besteht in der Eingabe der Werte von μ, σ^2 und p und der Auflösung für x:



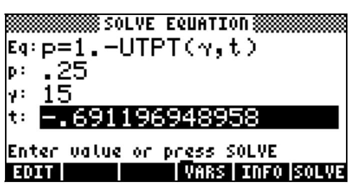
Dieses Eingabefeld kann zur Auflösung jeder der vier Variablen, die in der Gleichung für die Normalverteilung vorkommen, genutzt werden werden.

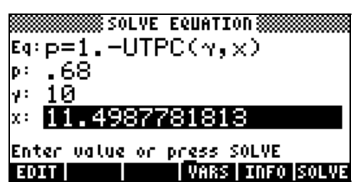
Um die Auflösung der Gleichungen mit den Funktionen UTPN, UTPT, UTPC und UTPF zu erleichtern, können Sie ein Unterverzeichnis UTPEQ anlegen, wo Sie die oben aufgelisteten Gleichungen abspeichern:

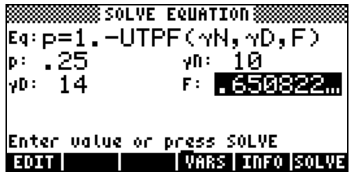




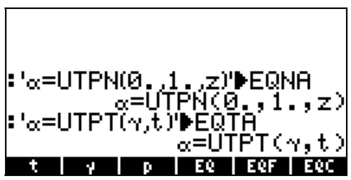
Daher haben Sie an diesem Punkt vier Gleichungen zur Auflösung zur Verfügung. Sie müssen nur eine der Gleichungen in das EQ-Feld in den numerischen Löser laden und mit der Lösung einer der variablen fortfahren. Beispiele für UTPT, UTPC und UPTF werden unten gezeigt:

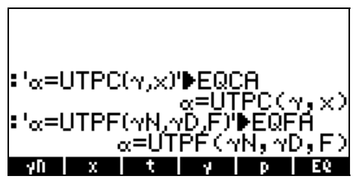




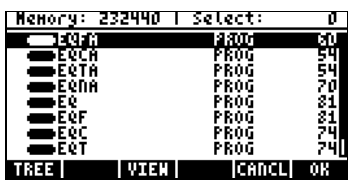


Wir weisen darauf hin, dass wir in allen oben gezeigten Beispielen mit p = P(X<x) arbeiten. Bei vielen statistischen Inferenzproblemen versuchen wie tatsächlich, den Wert von x zu finden, für den P(X>x) = α. Weiterhin arbeiten wir sehr am ehesten für die Normalverteilung mit Standardnormalverteilung, in welcher μ =0 und σ^2 = 1. Auf die Standardnormalvariable wird normalerweise als Z Bezug genommen, so dass das zu lösende Problem P(Z>z) = α ist. Für diese Fälle statistischer Inferenzproblemn können wir die folgenden Gleichungen speichern:

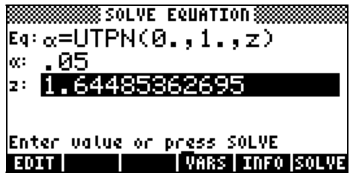


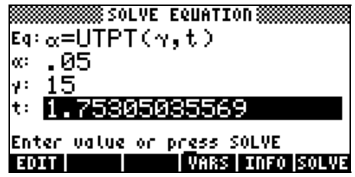


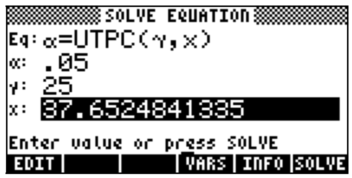
Mit diesen vier Gleichungen haben Sie, immer wenn Sie den numerischen Löser starten, die folgenden Auswahlmöglichkeiten:

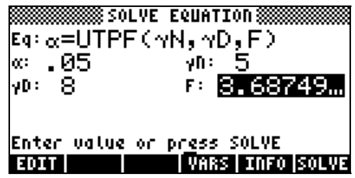


Beispiele für Lösungen der Gleichungen EQNA, EQTA, EQCA und EQFA werden unten gezeigt:


SOLVE EQUATION
Eq: α=UTPN(0.,1.,z)
α: .05
z: 1.64485362695
Enter value or press SOLVE
EDIT VARS INFO SOLVE


SOLVE EQUATION
Eq: α=UTPT(γ,t)
α: .05
γ: 15
t: 1.75305035569
Enter value or press SOLVE
EDIT VARS INFO SOLVE


SOLVE EQUATION
Eq: α=UTPC(γ,x)
α: .05
γ: 25
x: 37.6524841335
Enter value or press SOLVE
EDIT VARS INFO SOLVE

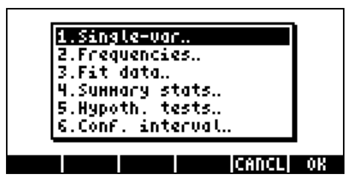

SOLVE EQUATION
Eq: α=UTPF(γN,γD,F)
α: .05
γN: 5
γD: 8
F: 3.68749…
Enter value or press SOLVE
EDIT VARS INFO SOLVE

Kapitel 18
Statistikanwendungen

In diesem Kapitel werden statistische Anwendungen des Taschenrechners vorgestellt, z. B. Stichprobenkenngrößen, Häufigkeitsverteilung von Daten, einfache Regression, Vertrauensbereiche und Hypothesentests.

Vorprogrammierte Statistikfunktionen

Der Taschenrechner enthält vorprogrammierte Statistikfunktionen, auf die über die Tastenkombination ↱ STAT zugegriffen werden kann (mit der Taste für die Zahl 5 identisch). Folgende Statistikanwendungen können mit dem Taschenrechner aufgerufen werden:



Die Anwendungen werden in diesem Kapitel ausführlich dargestellt. Zuerst jedoch zeigen wir, wie Daten für die statistische Analyse eingegeben werden.

Eingeben von Daten

Für die Analyse eines einzelnen Satzes von Daten (Stichprobe) können wir die Anwendungen 1, 2 und 4 in obiger Liste verwenden. Für alle diese Anwendungen ist es erforderlich, dass die Daten als Spalten der Matrix ΣDAT verfügbar sind. Sie können hierzu die Daten mithilfe von MatrixWriter (↰ MTRW) in Spalten eingeben.

Diese Vorgehensweise ist bei einer großen Anzahl von Datenpunkten möglicherweise ermüdend. Es bietet sich an, stattdessen die Daten als Liste (siehe Kapitel 8) einzugeben und die Liste mithilfe des Programms CRMC in einen Spaltenvektor zu konvertieren (siehe Kapitel 10). Stattdessen können Sie auch das folgende Programm eingeben, um eine Liste in einen Spaltenvektor zu konvertieren. Geben Sie das Programm im RPN-Modus ein:

« OBJ→ 1 2 →LIST →ARRY »

Speichern Sie das Programm in einer Variablen mit der Bezeichnung LCX. Nachdem Sie das Programm im RPN-Modus gespeichert haben, können Sie es auch im ALG-Modus verwenden.

Um einen Spaltenvektor in der Variablen ΣDAT zu speichern, verwenden Sie die Funktion STOΣ, die über den Katalog (‚N) verfügbar ist, z. B. STOΣ (ANS(1)) im ALG-Modus.

Beispiel 1 – Erstellen Sie mithilfe des oben definierten Programms LXC einen Spaltenvektor mit den folgenden Daten: 2.1 1.2 3.1 4.5 2.3 1.1 2.3 1.5 1.6 2.2 1.2 2.5.

Geben Sie im RPG-Modus die Daten in eine Liste ein:

{2.1 1.2 3.1 4.5 2.3 1.1 2.3 1.5 1.6 2.2 1.2 2.5 } ` @LXC

Speichern Sie die Daten mithilfe der Funktion STOΣ in ΣDAT.

Berechnen von Kenngrößen mit einer einzigen Variablen

Es wird vorausgesetzt, dass der einzelne Datensatz als Spaltenvektor in der Variablen ΣDAT gespeichert wurde. Drücken Sie ‚Ù, um auf die einzelnen Programme von STAT zuzugreifen. Drücken Sie @@@OK@@@, um **1. Single-var..** auszuwählen. Anschließend ist eine Eingabemaske mit der Beschriftung **SINGLE-VARIABLE STATISTICS** verfügbar und die derzeit in der Variablen ΣDAT vorhandenen Daten sind in der Maske als Vektor aufgelistet. Da nur eine Spalte vorhanden ist, muss vor dem Feld **Col:** der Wert 1 stehen. Das Feld **Type** bestimmt, ob Sie mit einer Stichprobe oder einer Grundgesamtheit arbeiten. Die Standardeinstellung ist Stichprobe. Bewegen Sie den Cursor an die horizontale Linie vor den Feldern **Mean**, **Std Dev**, **Variance**, **Total**, **Maximum** und **Minimum** und drücken Sie die Menütaste @✓CHK@, um die Werte auszuwählen, die von diesem Programm ausgegeben werden sollen. Drücken Sie anschließend @@@OK@@@. Die ausgewählten Werte werden mit der entsprechenden Beschriftung auf dem Bildschirm des Taschenrechners aufgelistet.

Beispiel 1 – Für die im vorherigen Beispiel gespeicherten Daten lauten die Ergebnisse der Kenngröße mit einer einzigen Variablen wie folgt:

Mean: 2.133,	Std Dev: 0.964,	Variance: 0.929
Total: 25.6,	Maximum: 4.5,	Minimum: 1.1

Definitionen

Für diese Größen gelten folgende Definitionen:

Angenommen, es sind mehrere Datenpunkte x_1, x_2, x_3, … vorhanden, die unterschiedliche Werte derselben diskreten oder kontinuierlichen Variablen X darstellen. Die Menge aller möglichen Werte der Größe x wird als Population von x bezeichnet. Eine endliche Population enthält nur eine bestimmte Anzahl der Elemente x_i. Wenn die Größe x den Wert einer kontinuierlichen Größe darstellt und daher theoretisch eine unendliche Anzahl von Werten annehmen kann, ist die Population in diesem Fall unendlich. Wenn Sie eine Teilmenge einer Grundgesamtheit auswählen, die durch n Datenwerte $\{x_1, x_2, \ldots, x_n\}$, dargestellt wird, haben Sie eine Stichprobe der Werte von x ausgewählt.

Stichproben sind durch eine Anzahl von Werten oder Kenngrößen gekennzeichnet. Es gibt Lagemaßzahlen, z. B. Mittelwerte, Medianwerte und häufigste Werte, sowie Streuungswerte, z. B. Wertebereich, Varianz und Standardabweichung.

Lagemaßzahlen

Der Mittelwert (bzw. das arithmetische Mittel) $\bar{x}$ der Stichprobe ist als Durchschnittswert der Elemente der Stichprobe definiert:

$$\bar{x} = \frac{1}{n} \cdot \sum_{i=1}^{n} x_i .$$

Dieser durch obige Gleichung ermittelte, mit `Total` beschriftete Wert stellt die Summe der Werte von x oder $\Sigma x_i = n\cdot \bar{x}$ dar. Dies ist der Wert, den der Taschenrechner unter der Überschrift `Mean` ausgibt. Andere in bestimmten Anwendungen verwendete Werte sind das geometrische Mittel x_g bzw. das harmonische Mittel x_h, die wie folgt definiert sind:

$$x_g = \sqrt[n]{x_1 \cdot x_2 \cdots x_n}, \qquad \frac{1}{x_h} = \sum_{i=1}^{n} \frac{1}{x_i}.$$

Beispiele für die Berechnung dieser Werte mithilfe von Listen finden Sie in Kapitel 8.

Der Median ist der Wert, der den Datensatz in der Mitte teilt, wenn die Elemente in aufsteigender Reihenfolge angeordnet sind. Bei einer ungeraden Zahl n von geordneten Elementen ist der Medianwert dieser Stichprobe der Wert an Position (n+1)/2. Bei einer geraden Zahl n von Elementen ist der Medianwert der Durchschnittswert der Elemente an den Positionen n/2 und (n+1)/2. Obwohl die vorprogrammierten Statistikfunktionen des Taschenrechners nicht die Berechnung des Medianwertes umfassen, kann auf sehr einfache Weise ein Programm geschrieben werden, mit dem unter Verwendung von Listen dieser Wert berechnet wird. Wenn Sie beispielsweise den Medianwert mithilfe der Daten in ΣDAT ermitteln möchten, geben Sie im RPN-Modus folgendes Programm ein (Weitere Informationen über das Programmieren in der Sprache User RPL finden Sie in Kapitel 21.):

```
« → nC « RCLΣ DUP SIZE 2 GET IF 1 > THEN nC COL– SWAP DROP
OBJ→ 1 + →ARRY END OBJ→ OBJ→ DROP DROP DUP → n « →LIST SORT
IF 'n MOD 2 == 0' THEN DUP 'n/2' EVAL GET SWAP '(n+1)/2' EVAL GET +
2 / ELSE '(n+1)/2' EVAL GET END "Median" →TAG » » »
```

Speichern Sie dieses Programm unter dem Namen MED. Es folgt ein Beispiel für die Anwendung dieses Programms.

Beispiel 2 – Um das Programm auszuführen, müssen Sie zunächst die Matrix ΣDAT vorbereiten. Geben Sie anschließend die Nummer der Spalte in ΣDAT ein, deren Medianwert Sie ermitteln möchten und drücken Sie MED. Verwenden Sie für die derzeit in ΣDAT vorhandenen Daten (die in einem vorherigen Beispiel eingegeben wurden) das Programm MED, um anzuzeigen, dass `Median: 2.15`.

Der <u>häufigste Wert</u> einer Stichprobe wird besser durch Histogramme bestimmt, daher wird er in einem späteren Abschnitt definiert.

Werte der Streubreite

Die <u>Varianz</u> (Var) der Stichprobe ist definiert als $s_x^2 = \frac{1}{n-1} \cdot \sum_{i=1}^{n} (x_i - \bar{x})^2$.

Die <u>Standardabweichung</u> (St Dev) der Stichprobe ist einfach die Quadratwurzel der Varianz, d. h. s_x.

Der <u>Wertebereich</u> der Stichprobe ist die Differenz zwischen Maximal- und Minimalwerten der Stichprobe. Da der Taschenrechner mit den vorprogrammierten Statistikfunktionen die Maximal- und Minimalwerte der Stichprobe ausgibt, können Sie den Wertebereich auf einfache Weise berechnen.

Variationskoeffizient

Beim Variationskoeffizienten einer Stichprobe ist der Mittelwert, eine Lagemaßzahl, mit der Standardabweichung, einem Wert der Streubreite, kombiniert und als Prozentzahl definiert durch: $V_x = (s_x/\ \bar{x})100$.

<u>Stichprobe und Grundgesamtheit</u>

Die oben verwendeten vorprogrammierten Funktionen für Kenngrößen mit einer einzigen Variablen können auf eine endliche Population angewendet werden, indem auf dem Bildschirm `SINGLE-VARIABLE STATISTICS` die Option `Type: Population` ausgewählt wird. Der Hauptunterschied besteht in den Werten der Varianz und der Standardabweichung, die berechnet werden, indem im Nenner der Varianz n und nicht (n-1) verwendet wird.

Beispiel 3 – Wenn Sie Beispiel 1 dieses Abschnitts wiederholt ausführen und als `Type` nicht `Sample`, sondern `Population` verwenden, erhalten Sie für Mittelwert, Gesamtwert, Maximum und Minimum dieselben Werte. Für Varianz und Standardabweichung sind jedoch folgende Werte gegeben: Varianz: 0.852, Standardabweichung: 0.923.

Erhalten von Häufigkeitsverteilungen

Die Anwendung **2. Frequencies** im Menü STAT kann zum Erhalten von Häufigkeitsverteilungen für einen Satz von Daten verwendet werden. Die Daten müssen wieder als Spaltenvektor verfügbar sein, der in der Variablen ΣDAT gespeichert ist. Drücken Sie zu Beginn ⬏ STAT ▼ OK. Die anschließend angezeigte Eingabemaske enthält die folgenden Felder:

ΣDAT:	die Matrix mit den betreffenden Daten.
Col:	die zu beachtende Spalte von ΣDAT.
X-Min:	die untere Klassengrenze (Standardwert = -6.5).
Bin Count:	die Anzahl der Klassen (Standardwert = 13).
Bin Width:	die einheitliche Breite jeder Klasse (Standardwert = 1).

Definitionen

Folgende Definitionen erleichtern das Verständnis der Bedeutung dieser Parameter: Wenn eine Menge von n Datenwerten $\{x_1, x_2, \ldots, x_n\}$ gegeben ist, die in keiner bestimmten Reihenfolge aufgelistet sind, ist es häufig erforderlich, diese Daten in einer Reihe von Klassen zu gruppieren, indem die Häufigkeit oder Anzahl der Werte jeder Klasse gezählt wird. (Hinweis: im Taschenrechner werden Klassen als "bins" = "Kästen" bezeichnet.)

Angenommen, die Klassen werden ermittelt, indem das Intervall (x_{bot}, x_{top}) in k = Bin Count Klassen unterteilt wird. Dies geschieht durch die Auswahl einer Anzahl von Klassengrenzen, also $\{xB_1, xB_2, \ldots, xB_{k+1}\}$, so dass Klasse 1 durch xB_1-xB_2 begrenzt ist, Klasse 2 durch xB_2- xB_3 usw. Die letzte Klasse, Klasse k, ist durch xB_k - xB_{k+1} begrenzt.

Der der Mitte jeder Klasse entsprechende Wert x wird als Klassenmittelpunkt bezeichnet und ist für i = 1, 2, …, k durch $xM_i = (xB_i + xB_{i+1})/2$ definiert.

Wenn die Klassen so gewählt werden, dass die Klassengröße identisch ist, können wir die Klassengröße als Bin Width = $\Delta x = (x_{max} - x_{min}) / k$ definieren,

und die Klassengrenzen können mit $xB_i = x_{bot} + (i - 1) * \Delta x$ berechnet werden.

Jeder Datenpunkt x_j, j = 1, 2, …, n gehört zur i-ten Klasse, wenn $xB_i \leq x_j < xB_{i+1}$.

Durch die Anwendung **2. Frequencies** im Menü STAT wird eine Häufigkeitszählung durchgeführt, und die Werte, die möglicherweise unter dem Minimum oder über dem Maximum der Klassengrenzen (d. h. die Ausreißer) liegen, werden protokolliert.

Beispiel 1 – Um das Ermitteln von Häufigkeitsverteilungen besser veranschaulichen zu können, möchten wir einen relativ umfangreichen Datensatz von 200 Datenpunkten erzeugen, indem wir wie folgt vorgehen:

- Zunächst aktivieren wir den Zufallszahlengenerator durch Verwendung von RDZ(25) im ALG-Modus oder 25 ENTER RDZ im RPN-Modus (siehe Kapitel 17).
- Geben Sie das folgende Programm im RPN-Modus ein:
  « → n « 1 n FOR j RAND 100 * 2 RND NEXT n →LIST » »
  und speichern Sie es unter dem Namen RDLIST (RanDom number LIST generator, Zufallszahlenlistengenerator).
- Generieren Sie die Liste von 200 Zahlen durch Verwendung von RDLIST(200) im ALG-Modus, oder von 200 ENTER RDLIST im RPN-Modus.
- Verwenden Sie das Programm LXC (siehe oben), um die auf diese Weise generierte Liste in einen Spaltenvektor zu konvertieren.
- Speichern Sie mithilfe der Funktion STOΣ den Spaltenvektor in ΣDAT.
- Ermitteln Sie über ↱ STAT OK Informationen zu den einzelne Variablen. Verwenden Sie als Typ des Datensatzes Sample und wählen Sie für die Ergebnisse alle Optionen aus. Die Ergebnisse für dieses Beispiel lauten:

Mean: 51.0406, Standardabweichung: .29.5893…, Variance: 875.529…
Total: 10208.12, Maximum: 99.35, Minimum: 0.13

Dies bedeutet, dass die Daten im Bereich von Werten nahe Null bis Werten nahe 100 liegen. Bei Verwendung ganzer Zahlen können wir den Bereich der Abweichung der Daten als (0,100) festlegen. Zum Erzeugen einer Häufigkeitsverteilung verwenden wir das Intervall (10,90) und unterteilen es in 8 Klassen mit der Breite 10.

- Wählen Sie das Programm **2. Frequencies** aus, indem Sie ↱ STAT ▼ OK drücken. Die Daten sind bereits in ΣDAT vorhanden und die Option Col muss den Wert 1 aufweisen, da ΣDAT nur eine einzige Spalte enthält.
- Ändern Sie X-Min in 10, Bin Count in 8 und Bin Width in 10 und drücken Sie OK.

Im RPN-Modus werden die Ergebnisse im Stack als Spaltenvektor auf Ebene 2 des Stacks angezeigt und als Zeilenvektor mit zwei Komponenten auf Ebene 1 des Stacks. Der Vektor auf Ebene 1 des Stacks stellt die Anzahl der Ausreißer außerhalb des Intervalls dar, für das die Häufigkeitszählung ausgeführt wurde. In diesem Fall erhalten Sie die Werte [25. 22.], die angeben, dass im ΣDAT-Vektor 25 Werte kleiner als 10 und 22 Werte größer als 90 vorhanden sind.

- Drücken Sie ◄, um die Vektorausreißer aus dem Stack zu entfernen. Das verbleibende Ergebnis ist die Häufigkeit der Daten. Dieses kann wie unten gezeigt in eine Tabelle übertragen werden.

Die Tabelle wurde anhand der Informationen erstellt, die wir zum Generieren der Häufigkeitsverteilung bereitgestellt hatten, obwohl die einzige vom Taschenrechner ausgegebene Spalte die Spalte f_i (Frequency, Häufigkeit) ist. Die Klassennummern und Klassengrenzen können bei Klassen einheitlicher Größe einfach berechnet werden, und der Klassenmittelpunkt ist einfach der Durchschnittswert der Klassengrenzen für jede Klasse. Schließlich wird die Summenhäufigkeit ermittelt, indem zu jedem Wert in der letzten Spalte außer dem ersten Wert die Häufigkeit in der nächsten Zeile addiert und das Ergebnis in der letzten Spalte der nächsten Zeile ersetzt wird. Somit ist die Summenhäufigkeit für die zweite Klasse 18+15 = 33, während die Summenhäufigkeit für die dritte Klasse 33+16 = 49 ist usw. Die

Summenhäufigkeit stellt die Häufigkeit der Zahlen dar, die kleiner oder gleich der oberen Grenze einer beliebigen Klasse sind.

Klassen-Nr. i	Klasse Grenze XB_i	XB_{i+1}	Klassenmittelpunkt Xm_i	Häufigkeit f_i	Summen-Häufigkeit
$< XB_1$	Ausreißer	unter Wertebereich		25	
1	10	20	15	18	18
2	20	30	25	14	32
3	30	40	35	17	49
4	40	50	45	17	66
5	50	60	55	22	88
6	60	70	65	22	110
7	70	80	75	24	134
k = 8	80	90	85	19	153
$>XB_k$	Ausreißer	über Wertebereich		22	

Wenn ein vom Taschenrechner generierter (Spalten-)Vektor der Häufigkeiten vorhanden ist, können Sie den Vektor der Summenhäufigkeit ermitteln, indem Sie im RPN-Modus folgendes Programm verwenden:

« DUP SIZE 1 GET → freq k « {k 1} 0 CON → cfreq « 'freq(1,1)' EVAL 'cfreq(1,1)' STO 2 k FOR j 'cfreq(j-1,1) +freq(j,1)' EVAL 'cfreq (j,1)' STO NEXT cfreq » » »

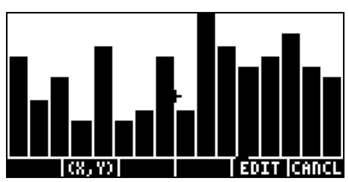
Speichern Sie das Programm unter dem Namen CFREQ. Verwenden Sie das Programm zum Generieren einer Liste von Summenhäufigkeiten (drücken Sie CFREQ, wenn der Spaltenvektor der Häufigkeiten im Stack vorhanden ist). Das Ergebnis für dieses Beispiel ist ein Spaltenvektor, der die letzte Spalte der obigen Tabelle darstellt.

Histogramme

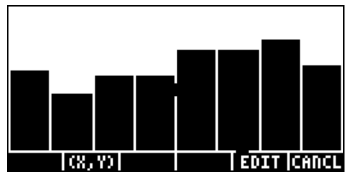
Ein Histogramm ist ein Balkendiagramm, in dem die Häufigkeit als Höhe der Balken und die Klassengrenzen als Sockel der Balken dargestellt werden. Wenn die Ursprungsdaten (d. h. die ursprünglichen Daten vor Ausführung der Häufigkeitszählung) in der Variablen ΣDAT vorhanden sind, können Sie als Diagrammtyp `Histogram` auswählen und den ursprünglichen Wert von x, die Anzahl der Klassen und die Klassenbreite angeben, um das Histogramm zu generieren. Sie können stattdessen auch wie im obigen Beispiel den Spaltenvektor generieren, der die Häufigkeitszählung enthält, diesen Vektor in ΣDAT speichern und als Diagrammtyp `Barplot` auswählen. Im nächsten Beispiel wird gezeigt, wie die erste Methode zum Generieren eines Histogramms verwendet wird.

Beispiel 1 – Generieren Sie unter Verwendung der im obigen Beispiel erzeugten 200 Datenpunkte (in ΣDAT als Spaltenvektor gespeichert) ein Histogramm der Daten, indem Sie X-Min = 10, Bin Count = 16 und Bin Width = 5 verwenden.

- Drücken Sie zunächst ← 2D/3D (gleichzeitig, sofern im RPN-Modus), um die Eingabe im Fenster PLOT SETUP zu aktivieren. Ändern Sie in diesem Fenster Type: in Histogramm und stellen Sie sicher, dass die Option Col: 1 ausgewählt ist. Drücken Sie anschließend NXT OK.
- Drücken Sie dann ← WIN (gleichzeitig, sofern im RPN-Modus), um die Eingabe im Fenster PLOT WINDOW – HISTOGRAM zu aktivieren. Ändern Sie die Informationen in diesem Fenster in H-View: 10 90, V-View: 0 15, Bar Width: 5.
- Drücken Sie ERASE DRAW, um das folgende Histogramm zu generieren:



- Drücken Sie CANCEL, um zum vorherigen Fenster zurückzukehren. Ändern Sie die Werte für V-View und Bar Width erneut, so dass diese nun wie folgt lauten: V-View: 0 30, Bar Width: 10. Das neue Histogramm, das auf demselben Datensatz beruht, wird nun wie folgt dargestellt:



Die Darstellung der Häufigkeit f_i gegen die Klassenmittelpunkte xM_i wird als Häufigkeitspolygon bezeichnet. Die Darstellung der Summenhäufigkeit gegen die oberen Grenzen wird als Häufigkeitsverteilungskurve der Summenhäufigkeit bezeichnet. Sie können Punktwolken erzeugen, die diese beiden Darstellungen simulieren, indem Sie die entsprechenden Daten in die Spalten 1 und 2 einer neuen ΣDAT-Matrix eingeben und im Fenster PLOT SETUP `Type`: in `SCATTER` ändern.

Anpassen von Daten an die Funktion y = f(x)

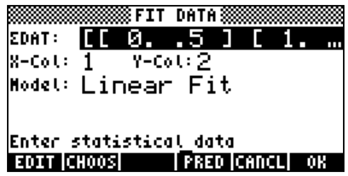
Mit dem Programm **3. Fit data**, das im Menü STAT als Option 3 verfügbar ist, können lineare, logarithmische, exponentielle und Potenzfunktionen an Datensätze (x,y) angepasst werden, die in Spalten der ΣDAT-Matrix gespeichert sind. Damit dieses Programm effektiv eingesetzt werden kann, müssen in der Variablen ΣDAT mindestens zwei Spalten vorhanden sein.

Beispiel 1 – Anpassen einer linearen Funktion an die Daten in der folgenden Tabelle:

x	0	1	2	3	4	5
y	0.5	2.3	3.6	6.7	7.2	11

- Geben Sie zunächst die Daten in den beiden Zeilen in die Spalten der Variablen ΣDAT ein, indem Sie MatrixWriter und die Funktion STOΣ verwenden.
- Verwenden Sie zum Aufrufen des Programms **3. Fit data** die folgende Tastenkombination: ↱ STAT ▼ ▼ OK. In der Eingabemaske wird die aktuelle Variable ΣDAT angezeigt, die bereits geladen ist. Legen Sie ggf.

für eine lineare Anpassung im Einrichtungsfenster die folgenden Parameter fest:



-
- Drücken Sie OK, um die Datenanpassung auszuführen. Die unten für unseren Datensatz dargestellte Ausgabe dieses Programms besteht im RPN-Modus aus den folgenden drei Zeilen:

3: '0.195238095238 + 2.00857142857*X'
2: Correlation: 0.983781424465
1: Covariance: 7.03

Auf Ebene 3 wird die Form der Gleichung dargestellt, in diesem Fall y = 0.06924 + 0.00383 x. Auf Ebene 2 wird der Stichprobenkorrelationskoeffizient der Stichprobe dargestellt, und auf Ebene 1 die Kovarianz von x-y.

Definitionen

Für eine Stichprobe von Datenpunkten (x,y) definieren wir die Stichprobenkovarianz als

$$s_{xy} = \frac{1}{n-1}\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})$$

Der Stichprobenkorrelationskoeffizient für x,y wird definiert als

$$r_{xy} = \frac{s_{xy}}{s_x \cdot s_y}.$$

Hierbei stellen s_x, s_y die Standardabweichungen von x bzw. y dar, d. h.

$$s_x^2 = \frac{1}{n-1}\sum_{i=1}^{n}(x_i - \overline{x})^2 \qquad s_y^2 = \frac{1}{n-1}\sum_{i=1}^{n}(y_i - \overline{y})^2$$

Bei den Werten s_{xy} und r_{xy} handelt es sich um die Werte für "Covariance" bzw. "Correlation", die mit der Funktion "Fit data" des Taschenrechners ermittelt wurden.

Linearisierte Funktionen

Zahlreiche gekrümmte Funktionen können zu einer linearen Form abgeflacht werden. Beispielsweise können die durch den Taschenrechner bereitgestellten unterschiedlichen Modelle für die Datenanpassung wie in der folgenden Tabelle dargestellt linearisiert werden.

Art der Anpassung	Tatsächliches Modell	Linearisiertes Modell	Unabh. Variable ξ	Abh. Variable η	Kovar. $s_{\xi\eta}$
Linear	y = a + bx	[dito]	x	y	s_{xy}
Log.	y = a + b ln(x)	[dito]	ln(x)	y	$s_{\ln(x),y}$
Exp.	$y = a\,e^{bx}$	ln(y) = ln(a) + bx	x	ln(y)	$s_{x,\ln(y)}$
Potenz.	$y = a\,x^b$	ln(y) = ln(a) + b ln(x)	ln(x)	ln(y)	$s_{\ln(x),\ln(y)}$

Die Stichprobenkovarianz von ξ,η ist durch $s_{\xi\eta} = \frac{1}{n-1}\sum(\xi_i - \overline{\xi})(\eta_i - \overline{\eta})$

definiert.

Wir definieren außerdem die Stichprobenvarianz von ξ bzw. η als

$$s_\xi^2 = \frac{1}{n-1}\sum_{i=1}^{n}(\xi_i - \overline{\xi})^2 \qquad s_\eta^2 = \frac{1}{n-1}\sum_{i=1}^{n}(\eta_i - \overline{\eta})^2$$

Der Stichprobenkorrelationskoeffizient $r_{\xi\eta}$ lautet $r_{\xi\eta} = \frac{s_{\xi\eta}}{s_{\xi} \cdot s_{\eta}}$.

Die allgemeine Form der Regressionsgleichung lautet $\eta = A + B\xi$.

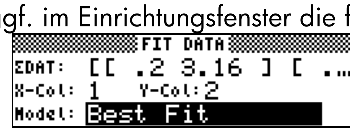
Optimale Datenanpassung

Der Taschenrechner kann bestimmen, welche der linearen oder linearisierten Funktionen die beste Anpassung für eine Menge von (x,y) Datenpunkten ergibt. Wir veranschaulichen die Verwendung dieser Funktion mit einem Beispiel. Angenommen, Sie möchten ermitteln, welche der Datenanpassungsfunktionen die beste Anpassung für die folgenden Daten ergibt:

x	0.2	0.5	1	1.5	2	4	5	10
y	3.16	2.73	2.12	1.65	1.29	0.47	0.29	0.01

Geben Sie zunächst die Daten als Matrix ein, indem Sie entweder die Daten unter Verwendung des Matrix-Editors eingeben oder mit dem in Kapitel 10 entwickelten Programm CRMC zwei Listen von Daten für x und y eingeben. Speichern Sie dann diese Matrix mit der Funktion STOΣ in der Statistikmatrix ΣDAT.

Starten Sie anschließend mit ↱ STAT ▼ ▼ OK die Anwendung zur Datenanpassung. Es wird die aktuelle Variable ΣDAT angezeigt, die bereits geladen ist. Legen Sie ggf. im Einrichtungsfenster die folgenden Parameter fest:



Drücken Sie OK, um die folgende Ausgabe zu erhalten:

1: '3.99504833324*EXP(-.579206831203*X)'
2: Correlation: -0.996624999526
3: Covariance: -6.23350666124

Die beste Anpassung für die Daten lautet daher $y = 3.995\ e^{-0.58 \cdot x}$.

Ermitteln zusätzlicher Summenkenngrößen

Für einige Berechnungen von Stichprobenkenngrößen bietet sich die Anwendung **4. Summary stats** im Menü STAT an. Drücken Sie zunächst erneut ↱ STAT, wechseln Sie mit der Taste mit dem Pfeil nach oben ▼ zur vierten Option und drücken Sie OK. Die anschließend angezeigte Eingabemaske enthält die folgenden Felder:

ΣDAT:	die Matrix mit den betreffenden Daten.
X-Col, Y-Col:	Die Verwendung dieser Optionen ist nur sinnvoll, wenn die Matrix ΣDAT mehr als zwei Spalten enthält. Standardmäßig wird für Spalte x die Spalte 1 und für Spalte y die Spalte 2 verwendet.
_Σ**X** _ Σ**Y…**:	Summenkenngrößen, die Sie als Ergebnisse dieses Programms auswählen können, indem Sie das entsprechende Feld mithilfe von [✓CHK] mit einem Häkchen versehen, wenn dieses Feld ausgewählt ist.

Viele dieser Summenkenngrößen werden zum Berechnen von Kenngrößen mit zwei Variablen (x,y) verwendet, die einen Bezug zur Funktion y = f(x) aufweisen. Daher gehört dieses Programm gewissermaßen zum Programm **3. Fit data**.

Beispiel 1 – Ermitteln Sie für die gegenwärtig in ΣDAT vorhandenen x-y-Daten alle Summenkenngrößen.

- Rufen Sie die Option **summary stats** mit ↱ STAT ▼ ▼ ▼ OK auf.
- Wählen Sie die den x- und y-Daten entsprechenden Spaltennummern aus, d. h. X-Col: 1 und Y-Col: 2.
- Wählen Sie mit der Taste ✓CHK alle Optionen für die Ausgabe aus, d. h. _ΣX, _ΣY usw.
- Drücken Sie OK, um die folgenden Ergebnisse zu erhalten:

  ΣX: 24.2, ΣY: 11.72, ΣX2: 148.54, ΣY2: 26.6246, ΣXY: 12.602, NΣ:8

> **Hinweis:** Das Menü STAT enthält zwei weitere Anwendungen, nämlich **5. Hypth. tests** und **6. Conf. Interval**. Diese beiden Anwendungen werden weiter unten in diesem Kapitel erläutert.

Berechnung von Perzentilen

Durch Perzentile wird ein Datensatz in 100 Teile unterteilt. Das grundlegende Verfahren zum Berechnen des 100 p-ten Perzentils (0 < p < 1) in einer Stichprobe der Größe n lautet wie folgt:

1. Ordnen Sie die n Größen in aufsteigender Reihenfolge an.
2. Bestimmen Sie das Produkt n·p.
   A. Wenn n·p keine Ganzzahl ist, runden Sie das Produkt auf die nächste Ganzzahl und ermitteln Sie den entsprechenden Wert in der Reihenfolge.
   B. Wenn n·p eine Ganzzahl ist, z. B. k, berechnen Sie den Mittelwert der k-th und (k-1) th geordneten Werte.

> **Hinweis**: Regel für das Runden auf Ganzzahlen: Wenn für einen nicht ganzzahligen Wert x.yz… der Wert y ≥ 5, runden Sie auf x+1. Wenn y < 5, runden Sie auf x.

Dieser Algorithmus kann durch das folgende, im RPN-Modus eingegebene Programm implementiert werden (Informationen über das Programmieren erhalten Sie in Kapitel 21):

```
« SORT DUP SIZE → p X n « n p * → k « IF k CEIL k FLOOR - NOT THEN X
k GET X k 1 + GET + 2 / ELSE k 0 RND X SWAP GET END » » »
```

Wir speichern dieses Programm in der Variablen %TILE (percent-tile bzw. Perzentil). Für dieses Programm ist als Eingabe ein Wert p zwischen 0 und 1 erforderlich, der das 100p-Perzentil darstellt, sowie eine Liste von Werten. Das Programm gibt das 100p-Perzentil der Liste zurück.

<u>Beispiel 1</u> – Bestimmen Sie das 37%-Perzentil der Liste { 2 1 0 1 3 5 1 2 3 6 7 9}. Geben Sie im RPN-Modus 0.27 `ENTER` { 2 1 0 1 3 5 1 2 3 6 7 9} `ENTER`

%TILE ein. Geben Sie im ALG-Modus %TILE(0.27,{2,1,0,1,3,5,1,2,3,6,7,9} ein. Das Ergebnis lautet 1.

Das Menü STAT

Über das Menü STAT kann auf sämtliche oben beschriebenen vorprogrammierten Statistikfunktionen zugegriffen werden. Sie können das Menü STAT aufrufen, indem Sie im RPN-Modus folgenden Befehl verwenden: 96 MENU

Sie können ein eigenes Programm erstellen, z. B. STAT, um das Menü STAT direkt zu aktivieren. Der Inhalt dieses Programms ist lediglich: « 96 MENU ».

Das Menü STAT enthält die folgenden Funktionen:

Wenn Sie eine der diesen Menüs entsprechenden Tasten drücken, erhalten Sie wie unten beschrieben Zugriff auf unterschiedliche Funktionen.

Das Untermenü DATA

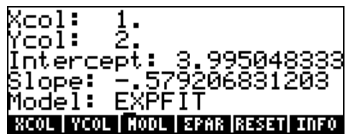
Das Untermenü DATA enthält Funktionen zum Bearbeiten der Statistikmatrix ΣDATA:

Diese Funktionen bewirken Folgendes:

Σ+ : fügt dem unteren Rand der Matrix ΣDATA eine Zeile auf Ebene 1 hinzu.
Σ- : entfernt die letzte Zeile in der Matrix ΣDATA und legt diese auf Ebene 1 des Stacks ab.
Die geänderte Matrix ΣDATA wird gespeichert.
CLΣ : löscht die aktuelle Matrix ΣDATA.
ΣDAT: legt Inhalte der aktuellen Matrix ΣDATA auf Ebene 1 des Stacks ab.
← ΣDAT: speichert die Matrix auf Ebene 1 des Stacks in der Matrix ΣDATA.

Das Untermenü ΣPAR

Das Untermenü ΣPAR enthält Funktionen zum Ändern von Statistikparametern. Die dargestellten Parameter entsprechen dem letzten Beispiel für die Datenanpassung.



Die auf dem Bildschirm angezeigten Parameter lauten:

Xcol: gibt die Spalte von ΣDATA an, die x darstellt (Standardwert: 1)
Ycol: gibt die Spalte von ΣDATA an, die y darstellt (Standardwert: 2)
Intercept: zeigt einen Abschnitt der letzten Datenanpassung an (Standardwert: 0)
Slope: zeigt die Steigung der letzten Datenanpassung an (Standardwert: 0)
Model: zeigt das aktuelle Datenanpassungsmodell an (Standardwert: LINFIT)

Die mit den Menütasten aufgerufenen Funktionen bewirken Folgendes:
XCOL: Aufruf mit n XCOL, ändert Spalte X in n.
YCOL: Aufruf mit n YCOL, ändert Spalte Y in n.
ΣPAR: zeigt Statistikparameter an.
RESET: setzt Parameter auf die Standardwerte zurück.
INFO: zeigt Statistikparameter an.

Das Untermenü MODL in Σ**PAR**
Dieses Untermenü enthält Funktionen, mit denen Sie durch Drücken der entsprechenden Taste das Datenanpassungsmodell in LINFIT, LOGFIT, EXPFIT, PWRFIT oder BESTFIT ändern können.

Das Untermenü 1VAR

Das Untermenü 1VAR enthält Funktionen zum Berechnen der Kenngrößen für die Spalten in der Matrix ΣDATA.

Folgende Funktionen sind verfügbar:

TOT : zeigt die Summe jeder Spalte in der Matrix ΣDATA an.
MEAN : zeigt den arithmetischen Mittelwert jeder Spalte in der Matrix ΣDATA an.
SDEV : zeigt die Standardabweichung jeder Spalte in der Matrix ΣDATA an.
MAXΣ : zeigt den Maximalwert jeder Spalte in der Matrix ΣDATA an.
MINΣ : zeigt den Minimalwert jeder Spalte in der Matrix ΣDATA an.
BINS : stellt bei Verwendung von x_s, Δx, n [BINS] die Häufigkeitsverteilung für Daten in der Spalte X der Matrix ΣDATA mit den durch $[x_s,x_s+\Delta x]$, $[x_s,x_s+2\Delta x]$,…, $[x_s,x_s+n\Delta x]$ definierten Häufigkeitsklassen bereit.
VAR : zeigt die Varianz jeder Spalte in der Matrix ΣDATA an.
PSDEV : zeigt die Populationsstandardabweichung (anhand von n statt von (n-1)) jeder Spalte in der Matrix ΣDATA an.
PVAR : zeigt die Populationsvarianz jeder Spalte in der Matrix ΣDATA an.
MINΣ : zeigt den Mindestwert jeder Spalte in der Matrix ΣDATA an.

Das Untermenü PLOT

Das Untermenü PLOT enthält Funktionen zum Erzeugen grafischer Darstellungen der Daten in der Matrix ΣDATA.

Diese Funktionen lauten:

BARPL : erzeugt ein Balkandiagramm mit den Daten in der Spalte Xcol der Matrix ΣDATA.
HISTP : erzeugt ein Histogramm der Daten in der Spalte Xcol der Matrix ΣDATA, wobei die 13 Klassen entsprechende Standardbreite verwendet wird, sofern die Klassengröße nicht mit der Funktion BINS im Untermenü 1VAR (siehe oben) geändert wird.

SCATR : erzeugt eine Punktwolke der Daten in der Spalte Ycol der Matrix ΣDATA gegen die Daten in der Spalte Xcol der Matrix ΣDATA. Die angepasste Gleichung wird in der Variablen EQ gespeichert.

Das Untermenü FIT

Das Untermenü FIT enthält Funktionen zum Anpassen von Gleichungen an die Daten in den Spalten Xcol und Ycol der Matrix ΣDATA.

Die in diesem Untermenü verfügbaren Funktionen lauten:

ΣLINE : stellt die der letzten Anpassung entsprechende Gleichung bereit.
LR : stellt Abschnitt und Steigung der letzten Anpassung bereit.
PREDX : Aufruf mit y PREDX, Ermitteln von x bei gegebenem y für die Anpassung y = f(x).
PREDY : Aufruf mit x PREDY, Ermitteln von y bei gegebenem x für die Anpassung y = f(x).
CORR : stellt den Korrelationskoeffizienten für die letzte Anpassung bereit.
COV : stellt die Stichprobenkovarianz für die letzte Anpassung bereit.
PCOV : zeigt die Populationskovarianz für die letzte Anpassung an.

Das Untermenü SUMS

Das Untermenü SUMS enthält Funktionen zum Ermitteln von Summenkenngrößen der Daten in den Spalten Xcol und Ycol der Matrix ΣDATA.

ΣX : stellt die Summe der Werte in der Spalte Xcol bereit.
ΣY : stellt die Summe der Werte in der Spalte Ycol bereit.
ΣX^2 : stellt die Summe der Quadrate der Werte in der Spalte Xcol bereit.
ΣY^2 : stellt die Summe der Quadrate der Werte in der Spalte Ycol bereit.
ΣX*Y : stellt die Summe von x·y bereit, d. h. der Produkte der Daten in den Spalten Xcol und Ycol.

NΣ : stellt die Anzahl der Spalten in der Matrix ΣDATA bereit.

Beispiel für Operationen des Menüs STAT

ΣDATA sei die auf der nächsten Seite dargestellte Matrix.

- Geben Sie mit dem Matrix-Editor die Matrix auf Ebene 1 des Stacks ein.
- Speichern Sie die Matrix in ΣDATA mit dem Befehl DATA ↰ ΣDAT
- Berechnen Sie die Kenngröße für jede Spalte: STAT 1VAR:

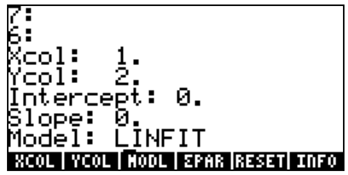
TOT	ergibt [38.5 87.5 82799.8]
MEAN	ergibt [5.5. 12.5 11828.54…]
SDEV	ergibt [3.39… 6.78… 21097.01…]
MAXΣ	ergibt [10 21.5 55066]
MINΣ	ergibt [1.1 3.7 7.8]
NXT VAR	ergibt [11.52 46.08 445084146.33]
PSDEV	ergibt [3.142… 6.284… 19532.04…]
PVAR	ergibt [9.87… 39.49… 381500696.85…]

- Daten:

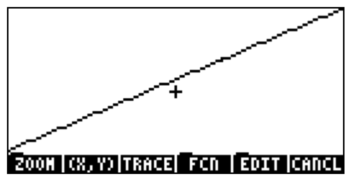
$$\begin{bmatrix} 1.1 & 3.7 & 7.8 \\ 3.7 & 8.9 & 101 \\ 2.2 & 5.9 & 25 \\ 5.5 & 12.5 & 612 \\ 6.8 & 15.1 & 2245 \\ 9.2 & 19.9 & 24743 \\ 10.0 & 21.5 & 55066 \end{bmatrix}$$

- Erzeugen Sie eine Punktwolke für die Daten in den Spalten 1 und 2 und zeichnen Sie eine entsprechende gerade Linie:

STAT ΣPAR RESET setzt die Statistikparameter zurück



NXT STAT PLOT SCATR	erzeugt Punktwolke
STATL	zeichnet die Datenanpassung als gerade Linie



CANCL	wechselt zum Hauptbildschirm

- Bestimmen Sie die Anpassungsgleichung und einige ihrer Kenngrößen:

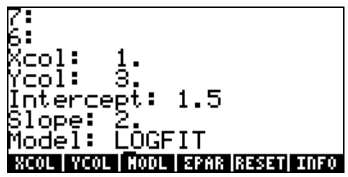
STAT FIT ΣLINE	ergibt `'1.5+2*X'`
LR	ergibt `Intercept: 1.5, Slope: 2`
3 PREDX	ergibt 0.75
1 PREDY	ergibt 3.50
CORR	ergibt 1.0
COV	ergibt 23.04
NXT PCOV	ergibt 19.74...

- Ermitteln Sie Summenkenngrößen für die Daten in den Spalten 1 und 2: STAT SUMS:

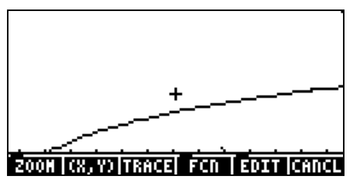
ΣX	ergibt 38.5
ΣY	ergibt 87.5
ΣX2	ergibt 280.87
ΣY2	ergibt 1370.23
ΣXY	ergibt 619.49
NΣ	ergibt 7

- Passen Sie die Daten in den Spalten 1 (x) und 3 (y) mit einer logarithmischen Anpassung an:

NXT STAT ΣPAR 3 YCOL	legt Ycol = 3 und
MODL LOGFI	legt Model = Logfit fest



NXT STAT PLOT SCATR	erzeugt ein Streuungsdiagramm von y gegen x
STATL	zeigt die Linie für logarithmische Anpassung an

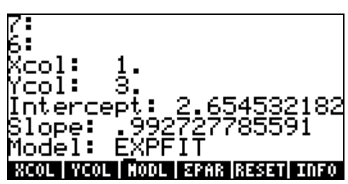


Offensichtlich ist die logarithmische Anpassung keine gute Lösung.

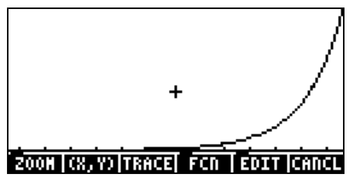
CANCL	wechselt zum Hauptbildschirm

- Wählen Sie mit folgendem Befehl die beste Anpassung aus:

STAT ΣPAR MODL BESTF	zeigt EXPFIT als beste Anpassung für diese Daten an



NXT STAT FIT ΣLINE	ergibt '2.6545*EXP(0.9927*X)'
CORR	ergibt 0.99995… (gute Korrelation)
2300 PREDX	ergibt 6.8139
5,2 PREDY	ergibt 463.37
NXT STAT PLOT SCATR	erzeugt ein Streuungsdiagramm von y gegen x
STATL	zeigt die Linie für logarithmische Anpassung an



- Um zum Menü STAT zurückzukehren, verwenden Sie NXT STAT
- Um wieder zum Variablenmenü zurückzukehren, verwenden Sie VAR.

Vertrauensbereiche

Bei der statistischen Inferenz handelt es sich um Schlussfolgerungen in Bezug auf eine Population anhand der aus den Stichprobendaten gewonnenen Informationen. Damit die Stichprobendaten aussagekräftig sind, muss die Stichprobe zufällig sein, d. h., die Auswahl einer bestimmten Stichprobe muss mit derselben Wahrscheinlichkeit wie die Auswahl jeder anderen möglichen Stichprobe aus einer bestimmten Population erfolgen. Im Folgenden werden einige für das Konzept der Zufallsstichprobe relevanten Begriffe erläutert:

- Population : die Menge aller denkbaren Werte eines Prozesses oder Attributs einer Komponente.
- Stichprobe : Teilmenge einer Population.
- Zufallsstichprobe : eine für die Population repräsentative Stichprobe.
- Zufallsvariable : für einen Stichprobenraum definierte reellwertige Funktion. Kann diskret oder kontinuierlich sein.

Falls eine bestimmte Wahrscheinlichkeitsverteilung auf die Population zutrifft, die von einem Parameter θ abhängt, kann zum Schätzen von θ eine Zufallsstichprobe von Werten (X_1, X_2, X_3, ..., X_n) der Größe n verwendet werden.

- Stichprobenverteilung : die gemeinsame Wahrscheinlichkeitsverteilung von $X_1, X_2, X_3, \ldots, X_n$.
- Kenngröße : jede Funktion der Werte, die quantifizierbar ist und keine unbekannten Parameter enthält. Eine Kenngröße ist eine Zufallsvariable, die für Schätzungen verwendet werden kann.
- Punktschätzung : wenn für den Parameter θ ein einziger Wert bereitgestellt wird.
- Vertrauensbereich : ein numerisches Intervall, das mit einer bestimmten Wahrscheinlichkeit den Parameter θ enthält.
- Schätzfunktion : Regel oder Methode zum Schätzen des Parameters θ.
- Schätzwert : der Wert, den die Schätzfunktion in einer bestimmten Anwendung zurückgibt.

Beispiel 1 – X stelle die Zeit (Stunden) dar, die für die Ausführung eines bestimmten Fertigungsprozesses erforderlich ist. Gegeben sei folgende Stichprobe der Werte von X: 2.2 2.5 2.1 2.3 2.2. Die Population, der die Stichprobe entnommen wurde, ist die Menge aller möglichen Werte für die Fertigungszeit und daher eine unendliche Population. Angenommen, der zu schätzende Populationsparameter ist der Mittelwert μ. Wir verwenden als Schätzfunktion den Mittelwert $\overline{X}$ der Stichprobe, der durch folgende Gleichung (Regel) definiert ist: $\overline{X} = \frac{1}{n} \cdot \sum_{i=1}^{n} X_i$.

Für die betreffende Stichprobe ist der Schätzwert für μ die Stichprobenkenngröße $\overline{x}$ = (2.2+2.5+2.1+2.3+2.2)/5 = 2.36. Dieser einzelne Wert von $\overline{X}$, also $\overline{x}$ = 2.36, stellt eine Punktschätzung des Populationsparameters μ dar.

Schätzung von Vertrauensbereichen

Die nächste Stufe der Inferenz nach der Punktschätzung ist die Intervallschätzung. Dies bedeutet, dass wir nicht einen einzelnen Wert einer

Schätzfunktion ermitteln, sondern zwei Kenngrößen a und b angeben, durch die ein Intervall definiert wird, das mit einer bestimmten Wahrscheinlichkeit den Parameter θ enthält. Die Endpunkte des Intervalls werden als Vertrauensgrenzen und das Intervall (a,b) als Vertrauensbereich bezeichnet.

Definitionen

(C_l,C_u) sei ein Vertrauensbereich, der den unbekannten Parameter θ enthält.

- Die statistische Sicherheit bzw. Aussagewahrscheinlichkeit ist die Menge $(1-\alpha)$ mit $0 < \alpha < 1$, so dass $P[C_l < \theta < C_u] = 1 - \alpha$, wobei P[ ] eine Wahrscheinlichkeit darstellt (siehe Kapitel 17). Der vorherige Ausdruck definiert die so genannten zweiseitigen Vertrauensgrenzen.
- Ein unterer einseitiger Vertrauensbereich wird durch $Pr[C_l < \theta] = 1 - \alpha$ definiert.
- Ein oberer einseitiger Vertrauensbereich wird durch $Pr[\theta < C_u] = 1 - \alpha$ definiert.
- Der Parameter α wird als Signifikanzniveau bezeichnet. Typische Werte für α sind 0.01, 0.05 und 0.1, die den Aussagewahrscheinlichkeiten 0.99, 0.95 bzw. 0.90 entsprechen.

Vertrauensbereiche für den Populationsmittelwert bei bekannter Populationsvarianz

$\bar{X}$ sei der Mittelwert einer Zufallsstichprobe der Größe n, die einer unendlichen Population mit der bekannten Standardabweichung σ entnommen wurde. Der zentrale zweiseitige Vertrauensbereich $100(1-\alpha)$% [d. h. 99 %, 95 %, 90 %, usw.] für den Populationsmittelwert μ ist $(\bar{X}-z_{\alpha/2}\cdot\sigma/\sqrt{n}\ ,\ \bar{X}+z_{\alpha/2}\cdot\sigma/\sqrt{n})$, wobei $z_{\alpha/2}$ eine normalverteilte Zufallsvariable darstellt, die mit der Wahrscheinlichkeit $\alpha/2$ überschritten wird. Der Standardfehler des Stichprobenmittelwertes $\bar{X}$ ist $\sigma/\sqrt{n}$.

Die obere und untere einseitige Vertrauensgrenze $100(1-\alpha)$% für den Populationsmittelwert μ lautet $X+z_\alpha\cdot\sigma/\sqrt{n}$ bzw. $\bar{X}-z_\alpha\cdot\sigma/\sqrt{n}$. Somit ist ein unterer einseitiger Vertrauensbereich durch $(-\infty\ ,\ X+z_\alpha\cdot\sigma/\sqrt{n})$ und ein oberer einseitiger Vertrauensbereich durch $(X-z_\alpha\cdot\sigma/\sqrt{n},+\infty)$ definiert. Beachten Sie, dass wir in diesen letzten beiden Vertrauensbereichen nicht den Wert $z_{\alpha/2}$, sondern z_α verwenden.

Im Allgemeinen ist der Wert z_k in der Standardnormalverteilung als der Wert von z definiert, dessen Überschreitungswahrscheinlichkeit k ist, d. h. $Pr[Z>z_k] = k$ oder $Pr[Z<z_k] = 1 - k$. Die Normalverteilung wurde in Kapitel 17 erläutert.

Vertrauensbereiche für den Populationsmittelwert bei unbekannter Populationsvarianz

$\bar{X}$ und S sei der Mittelwert bzw. die Standardabweichung einer Zufallsstichprobe der Größe n, die einer unendlichen Population mit Normalverteilung und der unbekannten Standardabweichung σ entnommen wurde. Der zentrale zweiseitige Vertrauensbereich 100(1-α)% [d. h. 99 %, 95 %, 90 %, usw.] für den Populationsmittelwert μ ist ($\bar{X} - t_{n-1,\alpha/2} \cdot S/\sqrt{n}$, $\bar{X} + t_{n-1,\alpha/2} \cdot S/\sqrt{n}$), wobei $t_{n-1,\alpha/2}$ eine Studentsche t-Verteilung mit dem Freiheitsgrad $\nu = n-1$ und der Überschreitungswahrscheinlichkeit $\alpha/2$ darstellt.

Die obere und untere einseitige Vertrauensgrenze 100· (1-α)% für den Populationsmittelwert μ lautet

$$X + t_{n-1,\alpha/2} \cdot S/\sqrt{n} \text{ bzw. } \bar{X} - t_{n-1,\alpha/2} \cdot S/\sqrt{n}.$$

Kleine und große Stichproben

Für die Studentsche t-Verteilung gilt, dass sie für n>30 nicht von der Standardnormalverteilung zu unterscheiden ist. Wenn daher bei Stichproben mit mehr als 30 Elementen die Populationsvarianz nicht bekannt ist, können Sie denselben Vertrauensbereich wie bei bekannter Populationsvarianz verwenden, müssen jedoch σ durch S ersetzen. Stichproben mit n>30 werden üblicherweise als große Stichproben bezeichnet, andernfalls als kleine Stichproben.

Vertrauensbereich für eine Quote

Eine diskrete Zufallsvariable X folgt einer Bernoulli-Verteilung, wenn X nur zwei Werte annehmen kann: X = 0 (Fehlschlag) und X = 1 (Erfolg). Wenn X ~ Bernoulli(p) und p die Erfolgswahrscheinlichkeit ist, ist der Mittelwert bzw. Erwartungswert von X gleich E[X] = p, und die Varianz lautet Var[X] = p(1-p).

Wenn ein Experiment mit X n Mal wiederholt wird und k erfolgreiche Ergebnisse aufgezeichnet werden, ist der Schätzwert p durch p' = k/n definiert, und der Standardfehler p' ist $\sigma_{p'} = \sqrt{(p\cdot(1-p)/n)}$. In der Praxis wird in der Formel für den Standardfehler p durch den Stichprobenschätzwert für p, d. h. p', ersetzt.

Bei einer großen Stichprobe mit n>30, n·p>5 und n·(1-p)>5 ist die Stichprobenverteilung nahezu eine Normalverteilung. Daher ist der zentrale zweiseitige Vertrauensbereich 100(1-α)% für den Populationsmittelwert $(p'+z_{\alpha/2}\cdot\sigma_{p'},\ p'+z_{\alpha/2}\cdot\sigma_p)$. Bei einer kleinen Stichprobe (n<30) kann mit $(p'-t_{n-1,\alpha/2}\cdot\sigma_{p'}, p'+t_{n-1,\alpha/2}\cdot\sigma_p)$ ein Schätzwert für den Vertrauensbereich ermittelt werden.

Stichprobenverteilung für Differenzen und Summen von Kenngrößen

S_1 und S_2 seien unabhängige Kenngrößen auf der Grundlage von zwei Stichproben der Größe n_1 bzw. n_2 aus zwei Populationen. Außerdem seien die jeweiligen Mittelwerte und Standardfehler der Stichprobenverteilungen dieser Kenngrößen μ_{S1} und μ_{S2} bzw. σ_{S1} und σ_{S2}. Die Differenz der Kenngrößen aus den beiden Populationen S_1-S_2 weist eine Stichprobenverteilung mit dem Mittelwert $\mu_{S1-S2} = \mu_{S1} - \mu_{S2}$ und dem Standardfehler $\sigma_{S1-S2} = (\sigma_{S1}^2 + \sigma_{S2}^2)^{1/2}$ auf. Außerdem weist die Summe der Kenngrößen T_1+T_2 den Mittelwert $_{S1+S2} = \mu_{S1} + \mu_{S2}$ und den Standardfehler $\sigma_{S1+S2} = (\sigma_{S1}^2 + \sigma_{S2}^2)^{1/2}$ auf.

Schätzfunktionen für den Mittelwert und die Standardabweichung von Differenz und Summe der Kenngrößen S_1 und S_2 sind durch folgende Gleichungen definiert:

$$\hat{\mu}_{S_1 \pm S_2} = \overline{X}_1 \pm \overline{X}_2, \qquad \hat{\sigma}_{S_1 \pm S_2} = \sqrt{\frac{\sigma_{S1}^2}{n_1} + \frac{\sigma_{S2}^2}{n_2}}$$

In diesen Ausdrücken sind $\overline{X}_1$ und $\overline{X}_2$ die Werte der Kenngrößen S_1 und S_2 von den aus den beiden Populationen entnommenen Stichproben, und σ_{S1}^2

und σ_{S2}^2 sind die Varianzen der Populationen der Kenngrößen S_1 und S_2 aus denen die Stichproben entnommen wurden.

Vertrauensbereiche für Summen und Differenzen von Mittelwerten

Wenn die Populationsvarianzen σ_1^2 und σ_2^2 bekannt sind, werden die Vertrauensbereiche für Differenz und Mittelwerte der Populationen, d. h. $\mu_1 \pm \mu_2$, durch folgenden Ausdruck definiert:

$$\left((\overline{X}_1 \pm X_2) - z_{\alpha/2} \cdot \sqrt{\frac{\sigma_1^2}{n_1} + \frac{\sigma_2^2}{n_2}}, (\overline{X}_1 \pm X_2) + z_{\alpha/2} \cdot \sqrt{\frac{\sigma_1^2}{n_1} + \frac{\sigma_2^2}{n_2}} \right)$$

Bei großen Stichproben, d. h. $n_1>30$ und $n_2>30$, und unbekannten, jedoch gleichen Populationsvarianzen $\sigma_1^2 = \sigma_2^2$ werden die Vertrauensbereiche für Differenz und Mittelwerte der Populationen, d. h. $\mu_1 \pm \mu_2$, durch folgenden Ausdruck definiert:

$$\left((\overline{X}_1 \pm X_2) - z_{\alpha/2} \cdot \sqrt{\frac{S_1^2}{n_1} + \frac{S_2^2}{n_2}}, (\overline{X}_1 \pm X_2) + z_{\alpha/2} \cdot \sqrt{\frac{S_1^2}{n_1} + \frac{S_2^2}{n_2}} \right).$$

Wenn eine der Stichproben klein ist, d. h. $n_1<30$ oder $n_2<30$, und die Populationsvarianzen $\sigma_1^2 = \sigma_2^2$ unbekannt, jedoch gleich sind, können wir für die Abweichung $\mu_1 \pm \mu_2$ den "zusammengefassten" Schätzwert $s_p^2 = [(n_1-1)\cdot s_1^2+(n_2-1)\cdot s_2^2]/(n_1+n_2-2)$ ermitteln.

In diesem Fall sind die zentralen Vertrauensbereiche für die Summe und Differenz der Mittelwerte der Populationen, d. h. $\mu_1 \pm \mu_2$, durch folgenden Ausdruck definiert:

$$\left((\overline{X}_1 \pm X_2) - t_{\nu,\alpha/2} \cdot s_p^2, (\overline{X}_1 \pm X_2) + t_{\nu,\alpha/2} \cdot s_p^2 \right)$$

Hierbei stellt $\nu = n_1+n_2-2$ den Freiheitsgrad in der Studentschen t-Verteilung dar.

Für die letzten beiden Optionen geben wir an, dass die Populationsvarianzen gleich sein müssen, obwohl sie nicht bekannt sind. Dies ist der Fall, wenn die beiden Stichproben derselben Population oder zwei Populationen entnommen wurden, von denen wir annehmen, dass sie dieselbe Populationsvarianz aufweisen. Wenn wir jedoch Grund zu der Annahme haben, dass die beiden unbekannten Populationsvarianzen voneinander abweichen, können wir folgenden Vertrauensbereich verwenden:

$$\left((\overline{X}_1 \pm X_2) - t_{\nu,\alpha/2} \cdot s^2_{\overline{X}_1 \pm \overline{X}_2},\ (\overline{X}_1 \pm X_2) + t_{\nu,\alpha/2} \cdot s^2_{\overline{X}_1 \pm \overline{X}_2}\right)$$

Hierbei ist die geschätzte Standardabweichung für Summe oder Differenz durch

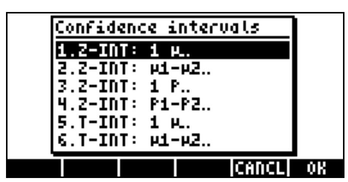
$$s_{\overline{X}_1 \pm \overline{X}_2} = \sqrt{\frac{s_1^2}{n_1} + \frac{s_2^2}{n_2}}$$

definiert und ν, der Freiheitsgrad der t-Abweichung, wird mit folgender Formel berechnet (das Ergebnis wird auf die nächste Ganzzahl gerundet):

$$\nu = \frac{[(S_1^2/n_1) + (S_2^2/n_2)]^2}{[(S_1^2/n_1)/(n_1-1)] + [(S_2^2/n_2)/(n_2-1)]}$$

Bestimmen von Vertrauensbereichen

Die Anwendung **6. Conf Interval** kann mit → STAT ▲ OK aufgerufen werden. Die Anwendung enthält die folgenden Optionen:

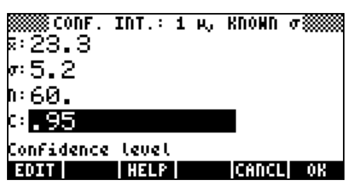


Diese Optionen werden im Folgenden erläutert:

1. Z-INT: 1 μ : Vertrauensbereich einer einzelnen Stichprobe für den Populationsmittelwert μ mit bekannter Populationsvarianz oder bei großen Stichproben mit unbekannter Populationsvarianz.
2. Z-INT: μ1–μ2: Vertrauensbereich für die Differenz der Populationsmittelwerte μ_1- μ_2 mit entweder bekannten Populationsvarianzen oder bei großen Stichproben mit unbekannten Populationsvarianzen.
3. Z-INT: 1 p : Vertrauensbereich einer einzelnen Stichprobe für die Quote p bei großen Stichproben mit unbekannter Populationsvarianz.
4. Z-INT: p1– p2: Vertrauensbereich für die Differenz zweier Quoten p_1-p_2 für große Stichproben mit unbekannten Populationsvarianzen.
5. T-INT: 1 μ : Vertrauensbereich einer einzelnen Stichprobe für den Populationsmittelwert μ bei kleinen Stichproben mit unbekannter Populationsvarianz.
6. T-INT: μ1–μ2: Vertrauensbereich für die Differenz zweier Populationsmittelwerte μ_1-μ_2 für kleine Stichproben mit unbekannten Populationsvarianzen.

Beispiel 1 – Bestimmen Sie den zentralen Vertrauensbereich für den Mittelwert einer Population, wenn eine Stichprobe mit 60 Elementen bedeutet, dass der Mittelwert der Stichprobe x = 23.2 und die Standardabweichung s = 5.2 beträgt. Verwenden Sie α = 0.05. Die statistische Sicherheit beträgt C = 1-α = 0.95.

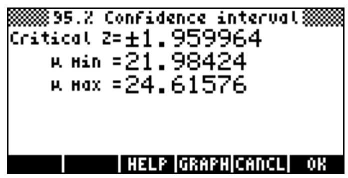
Wählen Sie die erste Option im oben abgebildeten Menü aus, indem Sie OK drücken. Geben Sie die erforderlichen Werte wie dargestellt in die Eingabemaske ein:



Drücken Sie HELP, um ein Fenster aufzurufen, in dem der Vertrauensbereich anhand vom Taschenrechner generierter Zufallszahlen erläutert wird. Blättern

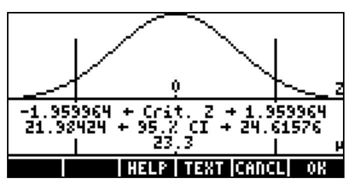
Sie im dann angezeigten Fenster mithilfe der Taste mit dem nach unten weisenden Pfeil ▽ nach unten. Drücken Sie OK, wenn Sie das Hilfefenster schließen möchten. Dadurch wird das oben abgebildete Fenster erneut angezeigt.

Drücken Sie OK, um den Vertrauensbereich zu berechnen. Das vom Taschenrechner angezeigte Ergebnis lautet:



Das Ergebnis bedeutet, dass ein Vertrauensbereich von 95 % berechnet wurde. Der im obigen Fenster angezeigte Wert für Critical z entspricht den Werten $\pm z_{\alpha/2}$ in der Formel des Vertrauensbereichs ($\bar{X} - z_{\alpha/2} \cdot \sigma/\sqrt{n}$, $\bar{X} + z_{\alpha/2} \cdot \sigma/\sqrt{n}$). Die Werte μ Min und μ Max stellen die obere bzw. untere Grenze dieses Intervalls dar, d. h μ Min = $\bar{X} - z_{\alpha/2} \cdot \sigma/\sqrt{n}$ und μ Max = $\bar{X} + z_{\alpha/2} \cdot \sigma/\sqrt{n}$.

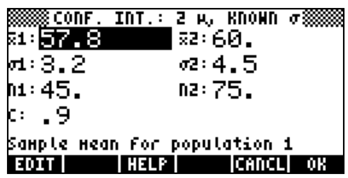
Drücken Sie GRAPH, um eine grafische Darstellung des Vertrauensbereichs anzuzeigen:



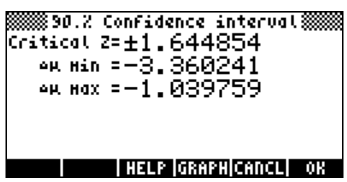
Das Diagramm stellt die Wahrscheinlichkeitsdichtefunktion, die Position der kritischen Punkte $\pm z_{\alpha/2}$, den Mittelwert (23.2) und die entsprechenden Bereichsgrenzen (21.88424 und 24.51576) dar. Drücken Sie TEXT, um zum vorherigen Ergebnisfenster zurückzukehren und/oder drücken Sie OK, um die Vertrauensbereichsumgebung zu schließen. Die Ergebnisse werden auf dem Bildschirm des Taschenrechners angezeigt.

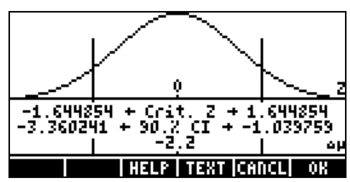
Beispiel 2 – Die Daten aus zwei Stichproben (Stichprobe 1 und 2) geben an, dass x_1 = 57.8 und $\bar{x}_2$ = 60.0. Die Stichprobengrößen betragen n_1 = 45 und n_2 = 75. Wenn bekannt ist, dass die Standardabweichungen der Populationen σ_1 = 3.2 und σ_2 = 4.5 betragen, bestimmen Sie den Vertrauensbereich von 90 % für die Differenz der Populationsmittelwerte, d. h. μ_1- μ_2.

Drücken Sie STAT OK, um die Vertrauensbereichsfunktion des Taschenrechners aufzurufen. Drücken Sie OK, um Option 2. Z-INT: μ1–μ2 auszuwählen. Geben Sie folgende Werte ein:



Drücken Sie anschließend OK. Die Ergebnisse werden unten in Text- und Diagrammform dargestellt:

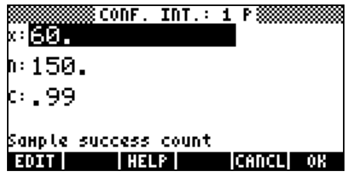




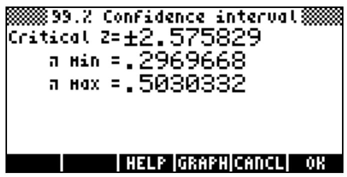
Die Variable Δμ stellt μ1–μ2 dar.

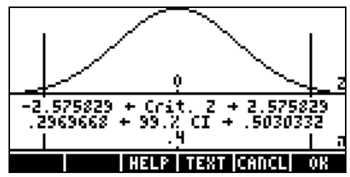
Beispiel 3 – Eine Meinungsumfrage gibt an, dass 60 Personen aus einer Stichprobe von 150 Personen höhere Vermögenssteuern zur Finanzierung öffentlicher Projekte befürworten. Bestimmen Sie einen Vertrauensbereich von 99 % für den Populationsanteil, der höhere Steuern befürwortet.

Drücken Sie STAT OK, um die Vertrauensbereichsfunktion des Taschenrechners aufzurufen. Drücken Sie OK, um Option 3. Z-INT: μ1–μ2 auszuwählen. Geben Sie folgende Werte ein:



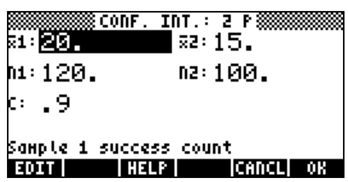
Drücken Sie anschließend OK. Die Ergebnisse werden unten als Text- und Diagramm dargestellt:



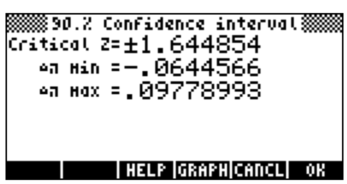


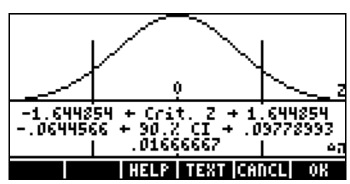
<u>Beispiel 4</u> – Bestimmen Sie einen Vertrauensbereich von 90 % für die Differenz der beiden Anteile, wenn Stichprobe 1 unter 120 Versuchen 20 Erfolge aufweist und Stichprobe 2 unter 100 Versuchen 15 Erfolge aufweist.

Drücken Sie ↱ STAT ▲ OK, um die Vertrauensbereichsfunktion des Taschenrechners aufzurufen. Drücken Sie ▼ ▼ ▼ OK, um Option 4. Z-INT: p1–p2 aufzurufen. Geben Sie folgende Werte ein:



Drücken Sie anschließend OK. Die Ergebnisse werden unten in Text- und Diagrammform dargestellt:

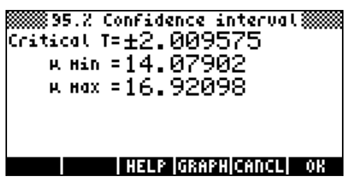


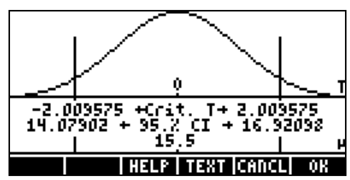


Beispiel 5 – Bestimmen Sie einen Vertrauensbereich von 95 % für den Mittelwert der Population, wenn eine Stichprobe von 50 Elementen den Mittelwert 15.5 und die Standardabweichung 5 aufweist. Die Standardabweichung der Population ist nicht bekannt.

Drücken Sie STAT, um die Vertrauensbereichsfunktion des Taschenrechners aufzurufen. Drücken Sie, um Option 5. T-INT: μ aufzurufen. Geben Sie folgende Werte ein:

Drücken Sie anschließend. Die Ergebnisse werden unten in Text- und Diagrammform dargestellt:

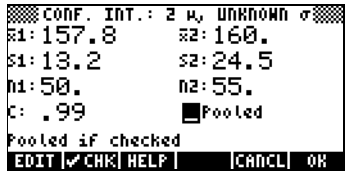




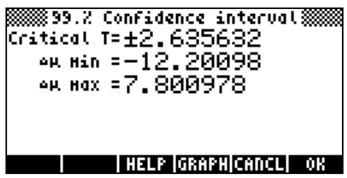
Die Abbildung stellt die Wahrscheinlichkeitsdichtefunktion der Studentschen t-Verteilung für den Freiheitsgrad ν = 50–1 = 49 dar.

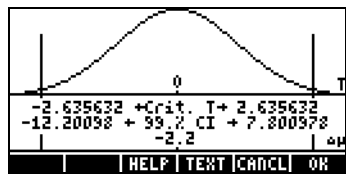
Beispiel 6 – Bestimmen Sie einen Vertrauensbereich von 99 % für die Differenz der Mittelwerte zweier Populationen mit den Stichprobendaten x_1 = 157.8, $\bar{x}_2$ = 160.0 und n_1 = 50, n_2 = 55. Die Standardabweichungen der Populationen betragen s_1 = 13.2 und s_2 = 24.5.

Drücken Sie STAT, um die Vertrauensbereichsfunktion des Taschenrechners aufzurufen. Drücken Sie, um Option 6. T-INT: μ1–μ2 aufzurufen. Geben Sie folgende Werte ein:

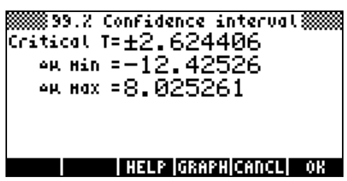


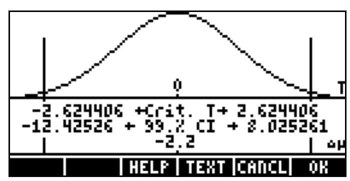
Drücken Sie anschließend OK. Die Ergebnisse werden unten in Text- und Diagrammform dargestellt:





Bei diesen Ergebnissen wird vorausgesetzt, dass die Werte s_1 und s_2 die Standardabweichungen der Populationen darstellen. Wenn diese Werte jedoch die Standardabweichungen der Stichproben darstellen, müssen Sie dieselben Werte wie zuvor, jedoch mit Auswahl der Option `_pooled` eingeben. Die Ergebnisse lauten nun:





Vertrauensbereiche für die Varianz

Zum Erstellen einer Formel für den Vertrauensbereich der Varianz stellen wir zuerst die Stichprobenverteilung der Varianz vor: Gegeben sei eine Zufallsstichprobe X_1, X_2 ..., X_n unabhängiger normalverteilter Variablen mit dem Mittelwert μ, der Varianz σ^2 und dem Stichprobenmittelwert $\overline{X}$. Die Kenngröße

$$\hat{S}^2 = \frac{1}{n-1} \cdot \sum_{i=1}^{n} (X_i - \overline{X})^2,$$

ist eine erwartungstreue Schätzfunktion der Varianz σ^2.

Die Menge $(n-1)\cdot\frac{\hat{S}^2}{\sigma^2}=\sum_{i=1}^{n}(X_i-\overline{X})^2$, weist eine Chi-Quadrat-Verteilung χ_{n-1}^2 mit dem Freiheitsgrad ν = n-1 auf. Der beidseitige Vertrauensbereich (1-α)·100 % wird durch

$$\Pr[\chi^2_{n-1,1-\alpha/2} < (n-1)\cdot S^2/\sigma^2 < \chi^2_{n-1,\alpha/2}] = 1-\alpha \text{ ermittelt.}$$

Der Vertrauensbereich für die Populationsvarianz σ^2 lautet daher

$$[(n-1)\cdot S^2/\ \chi^2_{n-1,\alpha/2}\,;\ (n-1)\cdot S^2/\ \chi^2_{n-1,1-\alpha/2}].$$

Hierbei stellen $\chi^2_{n-1,\alpha/2}$ und $\chi^2_{n-1,1-\alpha/2}$ die Werte dar, um die eine Variable χ^2 mit dem Freiheitsgrad ν = n-1 und den Überschreitungswahrscheinlichkeiten $\alpha/2$ bzw. 1- $\alpha/2$ von den Erwartungswerten abweichen kann.

Die obere einseitige Vertrauensbereichsgrenze für σ^2 ist durch $(n-1)\cdot S^2/\ \chi^2_{n-1,1-\alpha}$ definiert.

<u>Beispiel 1</u> – Bestimmen Sie anhand der Ergebnisse aus einer Stichprobe der Größe n = 25, die eine Stichprobenvarianz s^2 = 12.5 angibt, für die Populationsvarianz σ^2 einen Vertrauensbereich von 95 %.

In Kapitel 17 wird zum Lösen der Gleichung α = UTPC(γ,x) die numerische Lösung verwendet. In diesem Programm stellt γ den Freiheitsgrad (n-1) und α die Wahrscheinlichkeit für das Überschreiten eines bestimmten Wertes von x (χ^2) dar, d. h. $\Pr[\chi^2 > \chi_\alpha^2] = \alpha$.

Für das vorliegende Beispiel gilt α = 0.05, γ = 24 und α = 0.025. Die Lösung der oben dargestellten Gleichung lautet $\chi^2_{n-1,\alpha/2} = \chi^2_{24,0.025}$ = 39.3640770266.

Der Wert $\chi^2_{n-1,\alpha/2} = \chi^2_{24,0.975}$ wird hingegen anhand der Werte γ = 24 und α = 0.975 berechnet. Das Ergebnis lautet $\chi^2_{n-1,1-\alpha/2} = \chi^2_{24,0.975}$ = 12.4011502175.

Die oberen und unteren Grenzen des Vertrauensbereichs lauten wie folgt (führen Sie diese Berechnungen im ALG-Modus aus):

$$(n\text{-}1)\cdot S^2/\ \chi^2_{n-1,\alpha/2} = (25\text{-}1)\cdot 12.5/39.3640770266 = 7.62116179676$$

$$(n\text{-}1)\cdot S^2/\ \chi^2_{n-1,1-\alpha/2} = (25\text{-}1)\cdot 12.5/12.4011502175 = 24.1913044144$$

Der Vertrauensbereich von 95 % lautet für dieses Beispiel somit

$$7.62116179676 < \sigma^2 < 24.1913044144.$$

Hypothesentest

Eine Hypothese ist eine Aussage über eine Population (beispielsweise über ihren Mittelwert). Die Billigung dieser Aussage beruht auf einer statistischen Überprüfung einer der Population entnommenen Stichprobe. Der anschließende Vorgang und die anschließende Entscheidungsfindung werden als Hypothesentest bezeichnet.

Der Hypothesentest besteht aus dem Entnehmen einer Zufallsstichprobe aus der Population und dem Erstellen einer statistischen Hypothese über die Population. Wenn das postulierte Modell oder die postulierte Theorie durch die Werte nicht gestützt werden, wird die Hypothese verworfen. Wenn die Werte jedoch mit der Hypothese übereinstimmen, wird sie nicht verworfen, aber nicht notwendigerweise übernommen. Dieser Entscheidung ist ein Signifikanzniveau α zugeordnet.

Vorgehensweise beim Testen von Hypothesen

Die Vorgehensweise beim Hypothesentest besteht aus den folgenden sechs Schritten:

1. Geben Sie eine Nullhypothese H_0 an. Dies ist die zu testende Hypothese. Beispiel: H_0: μ_1-μ_2 = 0, d. h. wir nehmen an, dass die Mittelwerte von Population 1 und Population 2 identisch sind. Wenn H_0 wahr ist, wird jede Differenz der Mittelwerte Fehlern bei der Zufallsstichprobe zugeschrieben.
2. Geben Sie eine Alternativhypothese H_1 an. Diese kann für das vorliegende Beispiel H_1: μ_1-$\mu_2 \neq 0$ lauten. [Hinweis: Dies ist es, was wir eigentlich testen möchten.]

3. Bestimmen Sie eine Testkenngröße T, oder geben Sie diese an. Im vorliegenden Beispiel beruht T auf der Differenz der Mittelwerte $\bar{X}_1 - \bar{X}_2$.
4. Verwenden Sie die bekannte (oder vermutete) Verteilung der Testkenngröße T.
5. Definieren Sie anhand des zuvor zugewiesenen Signifikanzniveaus α einen Zurückweisungsbereich (die kritische Region R) für die Testkenngröße.
6. Bestimmen Sie anhand der ermittelten Daten, ob der berechnete Wert der Testkenngröße innerhalb oder außerhalb des kritischen Bereichs liegt. Wenn sich die Testkenngröße innerhalb des kritischen Bereichs befindet, sagen wir, dass die getestete Menge ein Signifikanzniveau von 100α Prozent aufweist.

Hinweise:

1. Für das vorliegende Beispiel ergibt die Alternativhypothese H_1: $\mu_1-\mu_2 \neq 0$ einen so genannten zweiseitigen Test. Wenn die Alternativhypothese H_1: $\mu_1-\mu_2 > 0$ oder H_1: $\mu_1-\mu_2 < 0$ lautet, liegt ein einseitiger Test vor.

2. Die Wahrscheinlichkeit des Zurückweisens der Nullhypothese ist gleich dem Signifikanzniveau, d. h. $\Pr[T\in R|H_0] = \alpha$. Die Notation $\Pr[A|B]$ stellt die bedingte Wahrscheinlichkeit von Ereignis A unter der Voraussetzung des Eintretens von Ereignis B dar.

Fehler beim Hypothesentest

Für Hypothesentests verwenden wir die Begriffe "Fehler vom Typ I" bzw. "Fehler vom Typ II", um Fälle zu definieren, in denen eine wahre Hypothese zurückgewiesen oder eine falsche Hypothese akzeptiert (nicht zurückgewiesen) wird. Es sei T = Wert der Testkenngröße, R = Zurückweisungsbereich, A = Beibehaltungsbereich, so dass $R\cap A = \varnothing$ und $R\cup A = \Omega$, wobei Ω = der Parameterraum für T und $\varnothing$ = die leere Menge ist. Die Wahrscheinlichkeiten für Fehler vom Typ I oder Typ II sind wie folgt definiert:

Zurückweisen einer wahren Hypothese $\Pr[\text{Fehler Typ I}] = \Pr[T\in R|H_0] = \alpha$
Nicht Zurückweisen einer falschen Hypothese $\Pr[\text{Fehler Typ II}] = \Pr[T\in A|H_1] = \beta$

Betrachten wir nun die Fälle, in denen wir die richtige Entscheidung treffen:

Nicht Zurückweisen einer wahren Hypothese $Pr[Not(Fehler\ Typ\ I)] = Pr[T \in A|H_0] = 1 - \alpha$

Zurückweisen einer falschen Hypothese $Pr[Not(Fehler\ Typ\ II)] = Pr\ [T \in R|H_1] = 1 - \beta$

Das Komplement von β wird als Mächtigkeit des Tests der Nullhypothese H_0 gegen die Alternativhypothese H_1 bezeichnet. Anhand der Mächtigkeit eines Tests wird beispielsweise die Mindestgröße einer Stichprobe bestimmt, um die Fehlerwahrscheinlichkeit zu verringern.

Auswählen der Werte von α und β

Ein typischer Wert des Signifikanzniveaus (oder der Wahrscheinlichkeit eines Fehlers vom Typ I) ist α = 0.05 (d. h. durchschnittlich eine falsche Zurückweisung pro 20 Tests). Wenn ein Fehler vom Typ I ernsthafte Folgen hat, wählen Sie für α kleinere Werte aus, z. B. 0.01 oder sogar 0.001.

Der Wert von β, d. h. die Wahrscheinlichkeit eines Fehlers vom Typ II, hängt von α, der Stichprobengröße n und dem tatsächlichen Wert des getesteten Parameters ab. Daher wird der Wert von β nach dem Ausführen des Hypothesentests bestimmt. Üblicherweise werden Diagramme gezeichnet, die β bzw. die Mächtigkeit des Tests (1- β) als Funktion des tatsächlichen Wertes des getesteten Parameters darstellen. Diese Diagramme werden als Operationscharakteristik bzw. Gütefunktion bezeichnet.

Inferenzen in Bezug auf einen einzigen Mittelwert

Zweiseitige Hypothese

Das Problem besteht im Testen der Nullhypothese H_o: $\mu = \mu_o$ gegen die Alternativhypothese H_1: $\mu \neq \mu_o$ bei einer statistischen Sicherheit von (1-α)100% oder einem Signifikanzniveau α bei einer Stichprobe der Größe n mit einem Mittelwert $\bar{x}$ und einer Standardabweichung s. Dieser Test wird als zweiseitiger Test bezeichnet. Der Test wird in folgenden Schritten ausgeführt:

Zunächst berechnen wir die entsprechende Kenngröße für den Test (t_o oder z_o) wie folgt:

- Wenn n < 30 und die Standardabweichung σ der Population bekannt ist, verwenden Sie die z-Kenngröße: $z_o = \frac{\bar{x} - \mu_o}{\sigma / \sqrt{n}}$
- Wenn n > 30 und σ bekannt ist, verwenden Sie z_o wie oben dargestellt. Wenn σ nicht bekannt ist, ersetzen Sie in z_o σ durch s, d. h. $z_o = \frac{\bar{x} - \mu_o}{s / \sqrt{n}}$.
- Wenn n < 30 und σ nicht bekannt ist, verwenden Sie die t-Kenngröße $t_o = \frac{\bar{x} - \mu_o}{s / \sqrt{n}}$, mit dem Freiheitsgrad ν = n - 1.

Berechnen Sie dann den entweder z_o oder t_o zugeordneten p-Wert (eine Wahrscheinlichkeit) und vergleichen Sie ihn mit α, um zu bestimmen, ob die Nullhypothese zurückgewiesen werden soll. Der p-Wert für einen zweiseitigen Test ist entweder durch

p-Wert = $P(|z|>|z_o|)$ oder durch p-Wert = $P(|t|>|t_o|)$ definiert.

Die beim Hypothesentest zu verwendenden Kriterien lauten:

- H_o zurückweisen, wenn p-Wert < α
- H_o nicht zurückweisen, wenn p-Wert > α

Der p-Wert für einen zweiseitigen Test kann mithilfe der Wahrscheinlichkeitsfunktionen des Taschenrechners wie folgt berechnet werden:

- Bei Verwendung von z: p-Wert = $2 \cdot$UTPN(0,1,$|z_o|$)
- Bei Verwendung von t: p-Wert = $2 \cdot$UTPT(ν,$|t_o|$)

Beispiel 1 – Testen Sie die Nullhypothese H_o: $\mu = 22.5$ ($= \mu_o$) gegen die Alternativhypothese H_1: $\mu \neq 22.5$ bei einer statistischen Sicherheit von 95 %, d. h. $\alpha = 0.05$, und verwenden Sie hierfür eine Stichprobe der Größe n = 25 mit dem Mittelwert $\bar{x} = 22.0$ und der Standardabweichung s = 3.5. Wir setzen voraus, dass wir den Wert der Populationsstandardabweichung nicht kennen. Daher berechnen wir die t-Kenngröße wie folgt:

$$t_o = \frac{\bar{x} - \mu_o}{s / \sqrt{n}} = \frac{22.0 - 22.5}{3.5 / \sqrt{25}} = -0.7142$$

Der entsprechende p-Wert für den Freiheitsgrad $\nu = 25 - 1 = 24$ lautet

p-Wert = 2·UTPT(24,-0.7142) = 2·0.7590 = 1.5169.

Da 1.5169 > 0.05, d. h. p-Wert > α, können wir die Nullhypothese H_o: μ = 22.0 nicht zurückweisen.

Einseitige Hypothese

Das Problem besteht im Testen der Nullhypothese H_o: $\mu = \mu_o$ gegen die Alternativhypothese H_1: $\mu > \mu_o$ oder H_1: $\mu < \mu_o$ bei einer statistischen Sicherheit von $(1-\alpha)100\%$ oder einem Signifikanzniveau α anhand einer Stichprobe der Größe n mit einem Mittelwert x und einer Standardabweichung s. Dieser Test wird als einseitiger Test bezeichnet. Die Ausführung eines einseitigen Tests beginnt wie der zweiseitige Test und wie oben dargestellt mit der Berechnung der entsprechenden Kenngröße für den Test (t_o oder z_o).

Anschließend berechnen wir den entweder z_o oder t_o zugeordneten p-Wert und vergleichen diesen mit α, um zu bestimmen, ob die Nullhypothese zurückgewiesen werden soll. Der p-Wert für einen zweiseitigen Test ist entweder definiert durch

p-Wert = $P(z > |z_o|)$ oder durch p-Wert = $P(t > |t_o|)$.

Die beim Hypothesentest zu verwendenden Kriterien lauten:

- H_o zurückweisen, wenn p-Wert < α

- H_o nicht zurückweisen, wenn p-Wert > α

Beachten Sie, dass es sich um dieselben Kriterien wie beim zweiseitigen Test handelt. Der Hauptunterschied liegt in der Art der Berechnung des p-Wertes. Der p-Wert für einen einseitigen Test kann mithilfe der Wahrscheinlichkeitsfunktionen des Taschenrechners wie folgt berechnet werden:

- Bei Verwendung von z: p-Wert = UTPN(0,1,z_o)
- Bei Verwendung von t : p-Wert = UTPT(ν,t_o)

Beispiel 2 – Testen Sie die Nullhypothese H_o: μ = 22,0 (= μ_o) gegen die Alternativhypothese H_1: μ >22.5 bei einer statistischen Sicherheit von 95 %, d. h. α = 0.05, und verwenden Sie hierfür eine Stichprobe der Größe n = 25 mit dem Mittelwert $\bar{x}$ = 22.0 und der Standardabweichung s = 3.5. Wir gehen wieder davon aus, dass wir den Wert der Populationsstandardabweichung nicht kennen. Daher ist der Wert der t-Kenngröße mit dem entsprechenden Wert des oben dargestellten zweiseitigen Tests identisch, d. h. t_o = -0.7142, und der p-Wert für Freiheitsgrad ν = 25 - 1 = 24 lautet

p-Wert = UTPT(24, |-0.7142|) = UTPT(24,0.7124) = 0.2409.

Da 0.2409 > 0.05, d. h. p-Wert > α, können wir die Nullhypothese H_o: μ = 22.0 nicht zurückweisen.

Inferenzen in Bezug auf zwei Mittelwerte

Die zu testende Nullhypothese lautet H_o: μ_1-μ_2 = δ bei einer statistischen Sicherheit von (1-α)100% oder dem Signifikanzniveau α und Verwendung zweier Stichproben mit den Größen n_1 und n_2, den Mittelwerten $\bar{x}_1$ und $\bar{x}_2$ sowie den Standardabweichungen s_1 und s_2. Wenn die den Stichproben entsprechenden Populationsstandardabweichungen σ_1 und σ_2 bekannt sind oder wenn n_1 > 30 und n_2 > 30 (große Stichproben), lautet die zu verwendende Testkenngröße

$$z_o = \frac{(\bar{x}_1 - \bar{x}_2) - \delta}{\sqrt{\frac{\sigma_1^2}{n_1} + \frac{\sigma_2^2}{n_2}}}$$

Wenn $n_1 < 30$ oder $n_2 < 30$ (mindestens eine kleine Stichprobe), verwenden Sie folgende Testkenngröße:

$$t = \frac{(\bar{x}_1 - \bar{x}_2) - \delta}{\sqrt{(n_1 - 1)s_1^2 + (n_2 - 1)s_2^2}} \sqrt{\frac{n_1 n_2 (n_1 + n_2 - 2)}{n_1 + n_2}}$$

<u>Zweiseitige Hypothese</u>

Wenn die Alternativhypothese eine zweiseitige Hypothese ist, d. h. H_1: μ_1-μ_2 $\neq \delta$, wird der p-Wert für diesen Test wie folgt berechnet:

- Bei Verwendung von z: p-Wert = 2·UTPN(0,1,|z_o|)
- Bei Verwendung von t : p-Wert = 2·UTPT(ν,|t_o|)

Hierbei wird der Freiheitsgrad der t-Verteilung durch $\nu = n_1 + n_2 - 2$ bestimmt. Die Testkriterien lauten

- H_o zurückweisen, wenn p-Wert < α
- H_o nicht zurückweisen, wenn p-Wert > α

<u>Einseitige Hypothese</u>

Wenn die Alternativhypothese eine zweiseitige Hypothese ist, d. h. H_1: μ_1-μ_2 < δ oder H_1: μ_1-μ_2 < δ, wird der p-Wert für diesen Test wie folgt berechnet:

- Bei Verwendung von z: p-Wert = UTPN(0,1, |z_o|)
- Bei Verwendung von t : p-Wert = UTPT(ν,|t_o|)

Die beim Hypothesentest zu verwendenden Kriterien lauten:

- H_o zurückweisen, wenn p-Wert < α
- H_o nicht zurückweisen, wenn p-Wert > α

Tests mit paarigen Stichproben

Wenn zwei Stichproben der Größe n mit paarigen Datenpunkten vorhanden sind, müssen wir diesen Fall wie eine einzige Stichprobe der Differenzen der paarigen Werte behandeln, statt die Nullhypothese H: μ_1-μ_2 = δ unter Verwendung der Mittelwerte und Standardabweichungen der beiden Stichproben zu testen. Mit anderen Worten, generieren Sie eine neue Zufallsvariable $X = X_1$-X_2 und testen sie H_o: $\mu = \delta$, wobei μ den Mittelwert der Population für X darstellt. Sie müssen daher $\bar{x}$ und s für die Stichprobe der Werte von x ermitteln. Der Test wird dann mit den bereits beschriebenen Methoden als Test mit einer einzigen Stichprobe fortgesetzt.

Inferenzen in Bezug auf eine einzige Quote

Angenommen, wir möchten die Nullhypothese H_0: $p = p_0$ testen, wobei p die Wahrscheinlichkeit eines erfolgreichen Ergebnisses bei einer beliebigen Wiederholung des Bernoulli-Versuchs darstellt. Zum Testen der Hypothese führen wir n Wiederholungen des Experiments durch und ermitteln, dass k erfolgreiche Ergebnisse aufgezeichnet werden. Somit wird durch p' = k/n ein Schätzwert von p angegeben.
Die Varianz der Abweichung wird mit $s_p^2 = p'(1\text{-}p')/n = k\cdot(n\text{-}k)/n^3$ geschätzt.

Angenommen, der Wert $Z = (p\text{-}p_0)/s_p$, entspricht der Standardnormalverteilung, d. h. Z ~ N(0,1). Der Wert der zu testenden Kenngröße lautet $z_0 = (p'\text{-}p_0)/s_p$.

Statt anhand des p-Wertes zu bestimmen, ob die Hypothese beibehalten wird, verwenden wir den Vergleich zwischen dem kritischen Wert von z0 und dem α oder $\alpha/2$ entsprechenden Wert von z.

<u>Zweiseitiger Test</u>

Bei Verwendung eines zweiseitigen Tests ermitteln wir den Wert von $z_{\alpha/2}$, mit

$$\Pr[Z> z_{\alpha/2}] = 1\text{-}\Phi(z_{\alpha/2}) = \alpha/2 \text{ oder } \Phi(z_{\alpha/2}) = 1\text{-}\ \alpha/2,$$

wobei $\Phi(z)$ die Summenverteilungsfunktion der Standardnormalverteilung darstellt (siehe Kapitel 17).

Weisen Sie die Nullhypothese H_0 zurück, wenn $z_0 > z_{\alpha/2}$ oder wenn $z_0 < - z_{\alpha/2}$.

Mit anderen Worten, der Zurückweisungsbereich ist $R = \{ |z_0| > z_{\alpha/2} \}$ und der Beibehaltungsbereich ist $A = \{|z_0| < z_{\alpha/2} \}$.

Einseitiger Test

Bei Verwendung eines einseitigen Tests ermitteln wir den Wert von S mit

$$\Pr[Z> z_\alpha] = 1-\Phi(z_\alpha) = \alpha \text{ oder } \Phi(z_\alpha) = 1- \alpha.$$

Weisen Sie die Nullhypothese H_0 zurück, wenn $z_0 > z_\alpha$ und H_1: $p>p_0$ oder wenn $z_0 < - z_\alpha$ und H_1: $p<p_0$.

Testen der Differenz zweier Quoten

Angenommen, wir möchten die Nullhypothese H_0: $p_1-p_2 = p_0$ testen, wobei p für die beiden Populationen 1 und 2 die Wahrscheinlichkeit eines erfolgreichen Ergebnisses einer beliebigen Wiederholung des Bernoulli-Versuchs darstellt. Zum Testen der Hypothese führen wir für Population 1 n Wiederholungen des Experiments durch und ermitteln, dass k_1 erfolgreiche Ergebnisse aufgezeichnet werden. Außerdem ermitteln wir für n_2 Versuche in Stichprobe 2 k_2 erfolgreiche Ergebnisse. Die Schätzwerte p_1 und p_2 sind somit durch $p_1' = k_1/n_1$ bzw. $p_2' = k_2/n_2$ definiert.

Die Varianzen für die Stichproben werden geschätzt als

$$s_1^2 = p_1'(1-p_1')/n_1 = k_1\cdot(n_1-k_1)/n_1^3 \text{ bzw. } s_2^2 = p_2'(1-p_2')/n_2 = k_2\cdot(n_2-k_2)/n_2^3.$$

Die Varianz der Quotendifferenz wird mit $s_p^2 = s_1^2 + s_2^2$ geschätzt.

Angenommen, der Wert $Z = (p_1-p_2-p_0)/s_p$, entspricht der Standardnormalverteilung, d. h. $Z \sim N(0,1)$. Der Wert der zu testenden Kenngröße lautet $z_0 = (p_1'-p_2'-p_0)/s_p$.

Zweiseitiger Test

Bei Verwendung eines zweiseitigen Tests ermitteln wir den Wert von $z_{\alpha/2}$ mit

$$Pr[Z> z_{\alpha/2}] = 1-\Phi(z_{\alpha/2}) = \alpha/2 \text{ oder } \Phi(z_{\alpha/2}) = 1- \alpha/2,$$

wobei $\Phi(z)$ die Summenverteilungsfunktion der Standardnormalverteilung darstellt.

Weisen Sie die Nullhypothese H_0 zurück, wenn $z_0 > z_{\alpha/2}$ oder wenn $z_0 < - z_{\alpha/2}$.

Mit anderen Worten, der Zurückweisungsbereich ist $R = \{ |z_0| > z_{\alpha/2} \}$ und der Beibehaltungsbereich ist $A = \{|z_0| < z_{\alpha/2} \}$.

<u>Einseitiger Test</u>
Bei Verwendung eines einseitigen Tests ermitteln wir den Wert von z_α mit

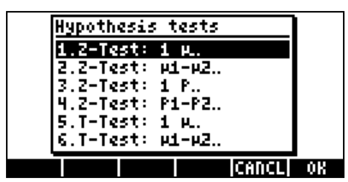
$$Pr[Z> z_\alpha] = 1-\Phi(z_\alpha) = \alpha \text{ oder } \Phi(z_\alpha) = 1- \alpha.$$

Weisen Sie die Nullhypothese H_0 zurück, wenn $z_0 > z_\alpha$ und H_1: $p_1-p_2 > p_0$ oder wenn $z_0 < - z_\alpha$ und H_1: $p_1-p_2 < p_0$.

Hypothesentest mit vorprogrammierten Funktionen

Der Taschenrechner enthält unter *5. Hypoth. tests* Hypothesentestprozeduren, die mit ⇨ STAT ▲ ▲ OK aufgerufen werden können.

Wie bei der bereits erläuterten Berechnung von Vertrauensbereichen bietet dieses Programm die folgenden 6 Optionen:



Diese Optionen werden entsprechend den Anwendungen für den Vertrauensbereich erläutert:

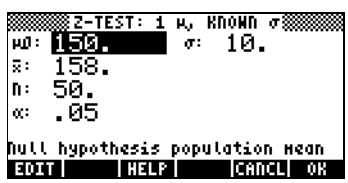
1. Z-Test: 1 μ: Hypothesentest einer einzelnen Stichprobe für den Populationsmittelwert μ mit bekannter Populationsvarianz oder bei großen Stichproben mit unbekannter Populationsvarianz.
2. Z-Test: μ1–μ2: Hypothesentest für die Differenz der Populationsmittelwerte μ_1- μ_2 mit entweder bekannten Populationsvarianzen oder bei großen Stichproben mit unbekannten Populationsvarianzen.
3. Z-Test: 1 p: Hypothesentest einer einzelnen Stichprobe für die Quote p bei großen Stichproben mit unbekannter Populationsvarianz.
4. Z-Test: p1– p2: Hypothesentest für die Differenz zweier Quoten p_1-p_2, bei großen Stichproben mit unbekannten Populationsvarianzen.
5. T-Test: 1 μ: Hypothesentest einer einzelnen Stichprobe für den Populationsmittelwert μ bei kleinen Stichproben mit unbekannter Populationsvarianz.
6. T-Test: μ1–μ2: Hypothesentest für die Differenz zweier Populationsmittelwerte μ_1-μ_2 bei kleinen Stichproben mit unbekannten Populationsvarianzen.

Führen Sie die folgenden Übungen aus:

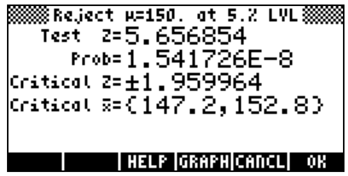
Beispiel 1 – Testen Sie für μ_0 = 150, σ = 10, $\bar{x}$ = 158, n = 50 und α = 0.05 die Hypothese H_0: μ = μ_o gegen die Alternativhypothese H_1: μ ≠ μ_0.

Drücken Sie ‚ STAT — — OK, um die Hypothesentestfunktion des Taschenrechners aufzurufen. Drücken Sie OK, um Option 1. Z-Test: 1 μ auszuwählen.

Geben Sie die folgenden Daten ein und drücken Sie OK:

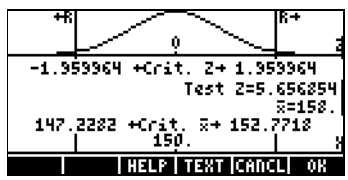


Anschließend werden Sie aufgefordert, die Alternativhypothese auszuwählen. Wählen Sie μ ≠150 aus und drücken Sie OK. Das Ergebnis lautet:



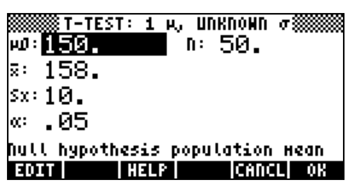
Anschließend weisen wir H$_0$: $\mu = 150$ gegen H$_1$: $\mu \neq 150$ zurück. Der Testwert z lautet z_0 = 5.656854. Der p-Wert lautet 1.54×10^{-8}. Die kritischen Werte $\pm z_{\alpha/2}$ = ±1.959964 entsprechen dem kritischen Bereich x {147.2 152.8}.

Diese Informationen können durch Drücken der Menütaste GRAPH grafisch dargestellt werden:

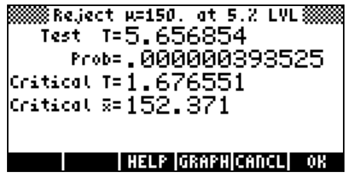


<u>Beispiel 2</u> – Testen Sie für μ_0 = 150, $\bar{x}$ = 158, s = 10, n = 50 und α = 0.05, die Hypothese H$_0$: $\mu = \mu_o$ gegen die Alternativhypothese H$_1$: $\mu > \mu_0$. Die Populationsstandardabweichung σ ist nicht bekannt.

Drücken Sie ↱ STAT ▲▲ OK, um die Hypothesentestfunktion des Taschenrechners aufzurufen. Drücken Sie ▲▲ OK, um Option 5. T-Test: 1 µ aufzurufen.
Geben Sie die folgenden Daten ein und drücken Sie OK:

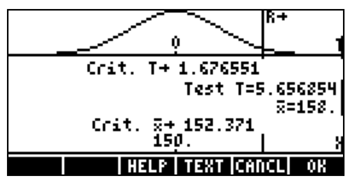


Wählen Sie die Alternativhypothese H$_1$: μ > 150 aus und drücken Sie OK. Das Ergebnis lautet:



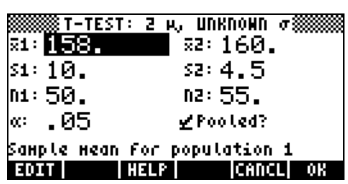
Wir weisen die Nullhypothese H_0: μ_0 = 150 gegen die Alternativhypothese H_1: μ > 150 zurück. Der Testwert lautet t_0 = 5.656854 mit p-Wert = 0.000000393525. Der kritische Wert von t lautet t_α = 1.676551 und entspricht dem kritischen Wert $\bar{x}$ = 152.371.

Drücken Sie GRAPH, um die Ergebnisse wie folgt grafisch darzustellen:

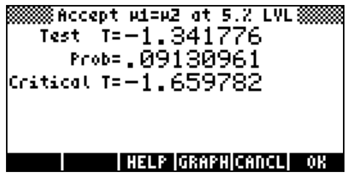


Beispiel 3 – Aufgrund der Daten zweier Stichproben gilt $\bar{x}_1$ = 158, $\bar{x}_1$ = 160, s_1 = 10, s_2 = 4.5, n1 = 50 und n_2 = 55. Testen Sie für α = 0.05 und eine "zusammengefasste" Varianz die Hypothese H_0: $\mu_1 - \mu_2 = 0$ gegen die Alternativhypothese H_1: $\mu_1 - \mu_2 < 0$.

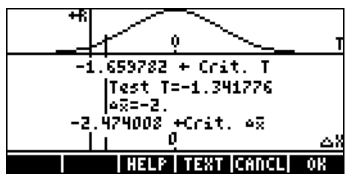
Drücken Sie → STAT ▲ ▲ OK, um die Hypothesentestfunktion des Taschenrechners aufzurufen. Drücken Sie ▲ OK, um Option 6. T-Test: $\mu 1 - \mu 2$ auszuwählen. Geben Sie die folgenden Daten ein und drücken Sie OK:



Wählen Sie die Alternativhypothese μ1< μ2 aus und drücken Sie OK. Das Ergebnis lautet



Somit behalten wir die Hypothese $\mu_1-\mu_2 = 0$ oder H_0: $\mu_1=\mu_2$ gegen die Alternativhypothese H_1: $\mu_1-\mu_2 < 0$ oder H_1: $\mu_1=\mu_2$ bei. Der Testwert t lautet t_0 = -1.341776 mit dem p-Wert = 0.09130961 und der kritische Wert t lautet $-t_\alpha$ = -.1659782. Die Ergebnisse werden wie folgt grafisch dargestellt:



Diese drei Beispiele sollten für das Verständnis der vorprogrammierten Hypothesentestfunktion des Taschenrechners ausreichen.

Inferenzen in Bezug auf eine einzige Varianz

Die zu testende Nullhypothese lautet H_o: $\sigma^2 = \sigma_o^2$ bei einer statistischen Sicherheit von $(1-\alpha)100\%$ oder dem Signifikanzniveau α sowie der Verwendung einer Stichprobengröße n und der Varianz s^2. Die zu verwendende Testkenngröße ist eine chi²-Testkenngröße, die wie folgt definiert ist:

$$\chi_o^2 = \frac{(n-1)s^2}{\sigma_0^2}$$

Je nach der ausgewählten Alternativhypothese wird der p-Wert wie folgt berechnet:

- H_1: $\sigma^2 < \sigma_o^2$ p-Wert = $P(\chi^2<\chi_o^2)$ = 1-UTPC(ν,χ_o^2)
- H_1: $\sigma^2 > \sigma_o^2$ p-Wert = $P(\chi^2>\chi_o^2)$ = UTPC(ν,χ_o^2)

- H_1: $\sigma^2 \neq \sigma_o^2$ p-Wert $=2\cdot\min[P(\chi^2<\chi_o^2), P(\chi^2>\chi_o^2)] =$
$2\cdot\min[1\text{-UTPC}(\nu,\chi_o^2), \text{UTPC}(\nu,\chi_o^2)]$

Hierbei erzeugt die Funktion min[x,y] den Minimalwert von x bzw. y (entsprechend erzeugt max[x,y] den Maximalwert von x bzw. y). UTPC(ν,x) stellt die oberen Wahrscheinlichkeiten des Taschenrechners für den Freiheitsgrad ν = n - 1 dar.

Die Testkriterien sind mit den für den Hypothesentest der Mittelwerte verwendeten Testkriterien identisch, also

- H_o zurückweisen, wenn p-Wert < α
- H_o nicht zurückweisen, wenn p-Wert > α

Beachten Sie, dass dieses Verfahren nur zulässig ist, wenn es sich bei der Population, aus der die Stichprobe entnommen wurde, um eine Normalpopulation handelt.

Besipiel 1 – Gegeben sei σ_o^2 = 25, α = 0.05, n = 25 und s^2 = 20, und die Stichprobe sei einer Normalpopulation entnommen. Zum Testen der Hypothese H_o: $\sigma^2 = \sigma_o^2$ gegen H_1: $\sigma^2 < \sigma_o^2$ berechnen wir zunächst

$$\chi_o^2 = \frac{(n-1)s^2}{\sigma_0^2} = \frac{(25-1)\cdot 20}{25} = 189.2$$

Mit dem Freiheitsgrad ν = n - 1 = 25 - 1 = 24 berechnen wir den p-Wert als

p-Wert = P(χ^2<19.2) = 1-UTPC(24,19.2) = 0.2587…

Da 0.2587… > 0.05, d. h. p-Wert > α, können wir die Nullhypothese H_o: σ^2 =25(= σ_o^2) nicht zurückweisen.

Inferenzen in Bezug auf zwei Varianzen

Die zu testende Nullhypothese lautet H_o: $\sigma_1^2 = \sigma_2^2$ bei einer statistischen Sicherheit von (1-α)100% oder dem Signifikanzniveau α sowie der Verwendung zweier Stichproben mit den Größen n_1 und n_2 und den

Varianzen s_1^2 und s_2^2. Die zu verwendende Testkenngröße ist eine Testkenngröße F, die wie folgt definiert ist:

$$F_o = \frac{s_N^2}{s_D^2}$$

Hierbei stellen s_N^2 und s_D^2 Zähler und Nenner der Kenngröße F dar. Die Auswahl von Zähler und Nenner hängt von der getesteten Alternativhypothese ab, wie unten dargestellt. Die entsprechende Verteilung für F weist die Freiheitsgrade $\nu_N = n_N-1$ und $\nu_D = n_D-1$ auf, wobei n_N und n_D die den Varianzen s_N^2 bzw. s_D^2 entsprechenden Stichprobengrößen darstellen.

In der folgenden Tabelle wird die Auswahl von Zähler und Nenner für F_o je nach der ausgewählten Alternativhypothese dargestellt:

Alternativ-hypothese	*Test-kenngröße*	*Freiheits-grad*
H_1: $\sigma_1^2 < \sigma_2^2$ (einseitig)	$F_o = s_2^2/s_1^2$	$\nu_N = n_2-1$, $\nu_D = n_1-1$
H_1: $\sigma_1^2 > \sigma_2^2$ (einseitig)	$F_o = s_1^2/s_2^2$	$\nu_N = n_1-1$, $\nu_D = n_2-1$
H_1: $\sigma_1^2 \neq \sigma_2^2$ (zweiseitig)	$F_o = s_M^2/s_m^2$ $s_M^2=\max(s_1^2,s_2^2)$, $s_m^2=\min(s_1^2,s_2^2)$	$\nu_N = n_M-1, \nu_D = n_m-1$

(*) n_M ist der s_M entsprechende Wert von n, und n_m ist der s_m entsprechende Wert von n.

Der p-Wert wird in sämtlichen Fällen wie folgt berechnet: p-Wert = $P(F>F_o)$ = UTPF(ν_N, ν_D,F_o)
Die Testkriterien lauten:

- H_o zurückweisen, wenn p-Wert < α
- H_o nicht zurückweisen, wenn p-Wert > α

<u>Beispiel 1</u> – Gegeben seien zwei Normalpopulationen entnommene Stichproben, so dass $n_1 = 21$, $n_2 = 31$, $s_1^2 = 0.36$ und $s_2^2 = 0.25$. Wir testen die Nullhypothese H_o: $\sigma_1^2 = \sigma_2^2$ bei Signifikanzniveau $\alpha = 0.05$ gegen die

Alternativhypothese H_1: $\sigma_1^2 \neq \sigma_2^2$. Für eine beidseitige Hypothese müssen wir s_M und s_m, wie folgt bestimmen:

$$s_M^2=\max(s_1^2,s_2^2) = \max(0.36,0.25) = 0.36 = s_1^2$$
$$s_m^2 = \min(s_1^2,s_2^2) = \max(0.36,0.25) = 0.25 = s_2^2$$

Außerdem gilt

$$n_M = n_1 = 21,$$
$$n_m = n_2 = 31,$$
$$\nu_N = n_M - 1= 21\text{-}1=20,$$
$$\nu_D = n_m -1 = 31\text{-}1 =30.$$

Die Testkenngröße F lautet daher $F_o = s_M^2/s_m^2=0.36/0.25=1.44$.

Der p-Wert lautet p-Wert = $P(F>F_o)$ = $P(F>1.44)$ = $UTPF(\nu_N, \nu_D,F_o)$ = $UTPF(20,30,1.44) = 0.1788\ldots$

Da $0.1788\ldots > 0.05$, d. h. p-Wert > α, können wir die Nullhypothese H_o: $\sigma_1^2 = \sigma_2^2$ nicht zurückweisen.

Weitere Anmerkungen zur linearen Regression

In diesem Kapitel werden die weiter oben in diesem Kapitel dargestellten Konzepte der linearen Regression weiter ausgearbeitet und ein Verfahren für den Hypothesentest von Regressionsparametern vorgestellt.

Die Methode der kleinsten Quadrate

x sei eine unabhängige nichtzufällige Variable und Y eine abhängige Zufallsvariable. Die Regressionskurve von Y auf x ist als die Beziehung zwischen x und dem Mittelwert der entsprechenden Verteilung der Werte von Y definiert.

Die Regressionskurve von Y auf x sei linear, d. h. die Mittelwertverteilung der Werte von Y ist durch $A + Bx$ definiert. Y unterscheidet sich vom Mittelwert ($A + B\cdot x$) durch den Wert ε, so dass

$Y = A + B\cdot x + \varepsilon$, wobei ε eine Zufallsvariable ist.

Zeichnen Sie ein Streuungsdiagramm oder eine Punktwolke, um visuell zu überprüfen, ob die Daten einem linearen Trend entsprechen.

Für n paarige Werte (x_i, y_i) sei y durch

$\hat{y} = a + b \cdot x$ definiert, wobei a und b konstant sind.

Der Prognosefehler sei als $e_i = y_i - \hat{y}_i = y_i - (a + b \cdot x_i)$ definiert.

Die Methode der kleinsten Quadrate erfordert, dass wir a und b so auswählen, dass die Summe quadratischer Fehler (SSE, Sum of Squared Errors) minimiert wird:

$$SSE = \sum_{i=1}^{n} e_i^2 = \sum_{i=1}^{n} [y_i - (a + bx_i)]^2$$

Die Bedingungen lauten:

$$\frac{\partial}{\partial a}(SSE) = 0 \qquad \frac{\partial}{\partial b}(SSE) = 0$$

Wir erhalten die so genannten Normalgleichungen:

$$\sum_{i=1}^{n} y_i = a \cdot n + b \cdot \sum_{i=1}^{n} x_i$$

$$\sum_{i=1}^{n} x_i \cdot y_i = a \cdot \sum_{i=1}^{n} x_i + b \cdot \sum_{i=1}^{n} x_i^2$$

Hierbei handelt es sich um ein lineares Gleichungssystem mit den Unbekannten a und b, das mit den Taschenrechnerfunktionen für lineare Gleichungen gelöst werden kann. Es ist jedoch nicht erforderlich, sich mit diesen Berechnungen zu beschäftigen, da Sie die weiter oben dargestellte Option **3. Fit Data** im Menü ↱ STAT verwenden können.

Hinweise:

- a,b sind erwartungstreue Schätzfunktionen von A, B.
- Das Gauß-Markov-Theorem besagt, dass unter allen erwartungstreuen Schätzfunktionen für A und B die Schätzfunktionen mit den kleinsten Quadraten (a,b) am effizientesten sind.

Weitere Gleichungen für lineare Regression

Die Summenkenngrößen, z. B. Σx, Σx^2 usw., können zum Definieren der folgenden Größen verwendet werden:

$$S_{xx} = \sum_{i=1}^{n}(x_i - \overline{x})^2 = (n-1)\cdot s_x^2 = \sum_{i=1}^{n} x_i^{\ 2} - \frac{1}{n}\left(\sum_{i=1}^{n} x_i\right)$$

$$S_y = \sum_{i=1}^{n}(y_i - \overline{y})^2 = (n-1)\cdot s_y^2 = \sum_{i=1}^{n} y_i^{\ 2} - \frac{1}{n}\left(\sum_{i=1}^{n} y_i\right)^2$$

$$S_{xy} = \sum_{i=1}^{n}(x_i - \overline{x})(y_i - \overline{y})^2 = (n-1)\cdot s_{xy} = \sum_{i=1}^{n} x_i y_i - \frac{1}{n}\left(\sum_{i=1}^{n} x_i\right)\left(\sum_{i=1}^{n} y_i\right)$$

Hieraus folgt, dass die Standardabweichungen von x und y sowie die Kovarianz von x,y durch

$s_x = \sqrt{\frac{S_{xx}}{n-1}}$, $s_y = \sqrt{\frac{S_{yy}}{n-1}}$ bzw. $s_{xy} = \frac{S_{yx}}{n-1}$ definiert sind.

Der Stichprobenkorrelationskoeffizient lautet $r_{xy} = \frac{S_{xy}}{\sqrt{S_{xx}\cdot S_{yy}}}$.

Für $\overline{x}$, $\overline{y}$, S_{xx}, S_{yy} und S_{xy} lautet die Lösung der Normalgleichungen

$$a = \overline{y} - b\overline{x} \ , \quad b = \frac{S_{xy}}{S_{xx}} = \frac{s_{xy}}{s_x^2}$$

Prognosefehler

Die Regressionskurve von Y auf x ist durch $Y = A + B \cdot x + \varepsilon$ definiert. Bei einer Menge von n Datenpunkten (x_i, y_i) gilt $Y_i = A + B \cdot x_i + \varepsilon_I$, (i = 1,2,…,n), mit Y_i = unabhängige normalverteilte Zufallsvariablen mit dem Mittelwert $(A + B \cdot x_i)$ und der gemeinsamen Varianz σ^2 sowie ε_i = unabhängige normalverteilte Zufallsvariablen mit dem Mittelwert Null und der gemeinsamen Varianz σ^2.

Es sei y_i = tatsächlicher Datenwert, $\hat{y}_i = a + b \cdot x_i$ = Datenprognose der kleinsten Quadrate. Der Prognosefehler lautet dann: $e_i = y_i - \hat{y}_i = y_i - (a + b \cdot x_i)$.

Ein Schätzwert von σ^2 ist der so genannte Standardfehler eines Schätzwertes

$$s_e^2 = \frac{1}{n-2}\sum_{i=1}^{n}[y_i - (a + bx_i)]^2 = \frac{S_{yy} - (S_{xy})^2 / S_{xx}}{n-2} = \frac{n-1}{n-2} \cdot s_y^2 \cdot (1 - r_{xy}^2)$$

Vertrauensbereiche und Hypothesentest bei linearer Regression

Im Folgenden sind einige Konzepte und Gleichungen für die statistische Inferenz bei linearer Regression aufgeführt:

- Vertrauensgrenzen für Regressionskoeffizienten:
  Für die Steigung (B): $b - (t_{n-2,\alpha/2}) \cdot s_e/\sqrt{S_{xx}} < B < b + (t_{n-2,\alpha/2}) \cdot s_e/\sqrt{S_{xx}}$,
  Für den Abschnitt (A):
  $a - (t_{n-2,\alpha/2}) \cdot s_e \cdot [(1/n) + \bar{x}^2/S_{xx}]^{1/2} < A < a + (t_{n-2,\alpha/2}) \cdot s_e \cdot [(1/n) + \bar{x}^2/S_{xx}]^{1/2}$,
  wobei t der Studentschen t-Verteilung mit dem Freiheitsgrad $\nu = n - 2$ entspricht und n die Anzahl der Datenpunkte in der Stichprobe darstellt.

- Hypothesentest für die Steigung B:
  Die Nullhypothese H_0: $B = B_0$, getestet gegen die Alternativhypothese H_1: $B \neq B_0$. Die Testkenngröße lautet $t_0 = (b - B_0)/(s_e/\sqrt{S_{xx}})$, wobei t der Studentschen t-Verteilung mit dem Freiheitsgrad $\nu = n - 2$ entspricht und n die Anzahl der Datenpunkte in der Stichprobe darstellt. Der Test wird wie ein Mittelwert-Hypothesentest ausgeführt, d. h, bei gegebenem

Signifikanzniveau α wird der kritische Wert von $t = t_{\alpha/2}$ bestimmt und anschließend H_0 zurückgewiesen, wenn $t_0 > t_{\alpha/2}$ oder wenn $t_0 < - t_{\alpha/2}$.

Wenn Sie den Test für den Wert $B_0 = 0$ ausführen und der Test ergibt, dass die Nullhypothese H_0: $B = 0$ nicht zurückgewiesen werden kann, ist die Gültigkeit der linearen Regression zweifelhaft. Mit anderen Worten, die Aussage $B \neq 0$ wird durch die Stichprobendaten nicht bestätigt. Es handelt sich daher um einen Test der Signifikanz des Regressionsmodells.

- Hypothesentest für die Steigung A:
  Die Nullhypothese H_0: $A = A_0$, getestet gegen die Alternativhypothese H_1: $A \neq A_0$. Die Testkenngröße lautet $t_0 = (a-A_0)/[(1/n)+ \bar{x}^2/S_{xx}]^{1/2}$, wobei t der Studentschen t-Verteilung mit dem Freiheitsgrad $\nu = n - 2$ entspricht und n die Anzahl der Datenpunkte in der Stichprobe darstellt. Der Test wird wie ein Mittelwert-Hypothesentest ausgeführt, d. h, bei gegebenem Signifikanzniveau α wird der kritische Wert von $t = t_{\alpha/2}$ bestimmt und anschließend H_0 zurückgewiesen, wenn $t_0 > t_{\alpha/2}$ oder wenn $t_0 < - t_{\alpha/2}$.

- Vertrauensbereich für den Mittelwert von Y bei $x = x_0$, i.e., $\alpha+\beta x_0$:
  $a+b\cdot x-(t_{n-2,\alpha/2})\cdot s_e\cdot[(1/n)+(x_0- \bar{x})^2/S_{xx}]^{1/2} < \alpha+\beta x_0 <$
  $a+b\cdot x+(t_{n-2, \alpha /2})\cdot s_e\cdot[(1/n)+(x_0- \bar{x})^2/S_{xx}]^{1/2}$.

- Prognosegrenzen: Vertrauensbereich für den Prognosewert $Y_0=Y(x_0)$:
  $a+b\cdot x-(t_{n-2,\alpha/2})\cdot s_e\cdot[1+(1/n)+(x_0- \bar{x})^2/S_{xx}]^{1/2} < Y_0 <$
  $a+b\cdot x+(t_{n-2, \alpha /2})\cdot s_e\cdot[1+(1/n)+(x_0- \bar{x})^2/S_{xx}]^{1/2}$.

Vorgehensweise mit dem Taschenrechner bei Inferenzkenngrößen für lineare Regression

1) Geben Sie (x,y) als Datenspalten in die Statistikmatrix ΣDAT ein.
2) Erzeugen Sie für die entsprechenden Spalten von ΣDAT eine Punktwolke und überprüfen Sie den linearen Verlauf anhand der entsprechenden Anzeige von H-VIEW und V-VIEW.
3) Verwenden Sie für die Datenanpassung als gerade Linie → STAT ▼ ▼ OK, und ermitteln Sie a, b, s_{xy} (Kovarianz) sowie r_{xy} (Korrelation).

4) Ermitteln Sie $\bar{x}$, $\bar{y}$, s_x, s_y mit ↱ STAT ▼ ,. In Spalte 1 werden die Kenngrößen für x und in Spalte 2 die Kenngrößen für y angezeigt.
5) Berechnen Sie

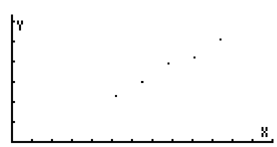
$$S_{xx} = (n-1) \cdot s_x^2, \quad s_e^2 = \frac{n-1}{n-2} \cdot s_y^2 \cdot (1 - r_{xy}^2)$$

6) Ermitteln Sie für Vertrauensbereiche oder zweiseitige Tests $t_{\alpha/2}$ mit dem Vertrauensbereich (1- α)100 % anhand einer t-Verteilung mit ν = n -2.
7) Ermitteln Sie für ein- oder zweiseitige Tests den Wert von t unter Verwendung der entsprechenden Gleichung für A oder B. Weisen Sie die Nullhypothese zurück, wenn `P-value` < α.
8) Verwenden Sie für Vertrauensbereiche die entsprechenden, oben dargestellten Formeln.

Beispiel 1 – Bestimmen Sie für die folgenden Daten (x,y) den Vertrauensbereich von 95 % für die Steigung B und den Abschnitt A.

x	2.0	2.5	3.0	3.5	4.0
y	5.5	7.2	9.4	10.0	12.2

Geben Sie die Daten (x,y) in die Spalten 1 bzw. 2 von ΣDAT ein. Eine Punktwolke der Daten veranschaulicht einen annähernd linearen Verlauf:



Verwenden Sie die Option `Fit Data` des Menüs ↱ STAT , um folgende Werte zu erhalten:

```
3: '-.86 + 3.24*X'
2: Correlation: 0.989720229749
1: Covariance: 2.025
```

Diese Ergebnisse bedeuten, dass a = -0.86, b = 3.24, r_{xy} = 0.989720229749 und s_{xy} = 2.025. Der Korrelationskoeffizient ist nahe genug an 1.0, um den linearen Verlauf des Diagramms zu bestätigen.

Über die Option `Single-var` des Menüs ⟨→⟩ STAT erhalten wir $\bar{x}$ = 3, s_x = 0.790569415042, $\bar{y}$ = 8.86, s_y = 2.58804945857.

Anschließend berechnen wir für n = 5

$$S_{xx} = (n-1) \cdot s_x^2 = (5-1) \cdot 0.790569415042^2 = 2.5$$

$$s_e^2 = \frac{n-1}{n-2} \cdot s_y^2 \cdot (1 - r_{xy}^2) =$$

$$\frac{5-1}{5-2} \cdot 2.5880...^2 \cdot (1 - 0.9897...^2) = 0.1826...$$

Vertrauensbereiche für die Steigung (Β) und den Abschnitt (A):

- Zunächst erhalten wir $t_{n-2,\alpha/2} = t_{3,0.025}$ = 3.18244630528 (Informationen über ein Programm zur Lösung der Gleichung für $t_{\nu,\alpha}$ erhalten Sie in Kapitel 17):
- Anschließend berechnen wir die Größen

  $(t_{n-2,\alpha/2}) \cdot s_e/\sqrt{S_{xx}} = 3.182\ldots \cdot (0.1826\ldots/2.5)^{1/2} = 0.8602\ldots$

  $(t_{n-2,\alpha/2}) \cdot s_e \cdot [(1/n) + \bar{x}^2/S_{xx}]^{1/2} =$
  $3.1824\ldots \cdot \sqrt{0.1826\ldots} \cdot [(1/5)+3^2/2.5]^{1/2} = 2.65$

- Schließlich lautet der Vertrauensbereich von 95 % für die Steigung B:

  (-0.86-0.860242, -0.86+0.860242) = (-1.72, -0.00024217)

Der Vertrauensbereich von 95 % lautet für den Abschnitt A: (3.24-2.6514, 3.24+2.6514) = (0.58855,5.8914).

Beispiel 2 – Angenommen, die in Beispiel 1 verwendeten Daten für y stellen die Dehnung (in hundertstel Zentimeter) eines einer Kraft x (in zehntel Kilogramm) ausgesetzten Metallseiles dar. Wir nehmen für dieses physikalische Phänomen an, dass der Abschnitt A den Wert Null aufweist. Zur Überprüfung dieser Annahme testen wir die Nullhypothese H_0: $A = 0$ gegen die Alternativhypothese H_1: $A \neq 0$ bei dem Signifikanzniveau $\alpha = 0.05$.

Die Testkenngröße lautet $t_0 = (a\text{-}0)/[(1/n)+ \bar{x}^2/S_{xx}]^{1/2} = (-0.86)/[(1/5)+3^2/2.5]^{½} = -0.44117$. Der kritische Wert von t für $\nu = n - 2 = 3$ und $\alpha/2 = 0.025$ kann mit der in Kapitel 17 entwickelten numerischen Lösung für die Gleichung α = UTPT(γ,t) berechnet werden. In diesem Programm stellt γ den Freiheitsgrad (n-2) und α die Wahrscheinlichkeit für das Überschreiten eines bestimmten Wertes von t dar, d. h. $\Pr[\, t>t_\alpha] = 1 - \alpha$. Im vorliegenden Beispiel lautet der Wert des Signifikanzniveaus $\alpha = 0.05$, g = 3, und $t_{n\text{-}2,\alpha/2} = t_{3,0.025}$. Zudem gilt für $\gamma = 3$ und $\alpha = 0.025$, dass $t_{n\text{-}2,\alpha/2} = t_{3,0.025} = 3.18244630528$. Da $t_0 > - t_{n\text{-}2,\alpha/2}$, können wir die Nullhypothese H_0: $A = 0$ gegen die Alternativhypothese H_1: $A \neq 0$ bei dem Signifikanzniveau $\alpha = 0.05$. nicht zurückweisen.
Dieses Ergebnis bedeutet, dass A = 0 für die lineare Regression geeignet ist. Schließlich wurde für a der Wert –0.86 ermittelt. Dieser Wert ist relativ nahe bei Null.

Beispiel 3 – Testen Sie die Signifikanz für die lineare Regression. Testen Sie die Nullhypothese für die Steigung H_0: $B = 0$ gegen die Alternativhypothese H_1: $B \neq 0$ beim Signifikanzniveau $\alpha = 0.05$ für die lineare Anpassung von Beispiel 1.

Die Testkenngröße lautet $t_0 = (b\ \text{-}B_0)/(s_e/\sqrt{S_{xx}}) = (3.24\text{-}0)/(\sqrt{0.18266666667/2.5}) = 18.95$. Der kritische Wert von t für $\nu = n - 2 = 3$ und $\alpha/2 = 0.025$ wurde in Beispiel 2 als $t_{n\text{-}2,\alpha/2} = t_{3,0.025} = 3.18244630528$ ermittelt. Da $t_0 > t_{\alpha/2}$, müssen wir die Nullhypothese H_1: $B \neq 0$ beim Signifikanzniveau $\alpha = 0.05$ für die lineare Anpassung von Beispiel 1 zurückweisen.

Mehrfache lineare Anpassung

Gegeben sei ein Datensatz der Form

$\mathbf{x_1}$	$\mathbf{x_2}$	$\mathbf{x_3}$	**…**	$\mathbf{x_n}$	**y**
x_{11}	x_{21}	x_{31}	…	x_{n1}	y_1
x_{12}	x_{22}	x_{32}	…	x_{n2}	y_2
x_{13}	x_{32}	x_{33}	…	x_{n3}	y_3
.	.	.		.	.
.	.	.	.	.	.
$x_{1,m-1}$	$x_{2,m-1}$	$x_{3,m-1}$	…	$x_{n,m-1}$	y_{m-1}
$x_{1,m}$	$x_{2,m}$	$x_{3,m}$	…	$x_{n,m}$	y_m

Angenommen, wir möchten eine Datenanpassung der Form $y = b_0 + b_1 \cdot x_1 + b_2 \cdot x_2 + b_3 \cdot x_3 + \ldots + b_n \cdot x_n$ ermitteln. Sie können die Näherung der kleinsten Quadrate an die Werte der Koeffizienten **b** = $[b_0 \ \ b_1 \ \ b_2 \ \ b_3 \ \ldots \ b_n]$ durch Erzeugen der Matrix **X** erhalten:

$$\begin{bmatrix} 1 & x_{11} & x_{21} & x_{31} & \ldots & x_{n1} \\ 1 & x_{12} & x_{22} & x_{32} & \ldots & x_{n2} \\ 1 & x_{13} & x_{32} & x_{33} & \ldots & x_{n3} \\ . & . & . & . & & . \\ . & . & . & . & . & . \\ 1 & x_{1,m} & x_{2,m} & x_{3,m} & \ldots & x_{n,m} \end{bmatrix}$$

Der Vektor der Koeffizienten wird dann mit $\mathbf{b} = (\mathbf{X}^T \cdot \mathbf{X})^{-1} \cdot \mathbf{X}^T \cdot \mathbf{y}$ ermittelt, wobei **y** den Vektor $\mathbf{y} = [y_1 \, y_2 \ \ldots \ y_m]^T$ darstellt.

Verwenden Sie als *Beispiel* die folgenden Daten, um die mehrfache lineare Anpassung zu ermitteln.

$$y = b_0 + b_1 \cdot x_1 + b_2 \cdot x_2 + b_3 \cdot x_3,$$

x_1	x_2	x_3	y
1.20	3.10	2.00	5.70
2.50	3.10	2.50	8.20
3.50	4.50	2.50	5.00
4.00	4.50	3.00	8.20

6.00	5.00	3.50	9.50

Sie können mit dem Taschenrechner im RPN-Modus wie folgt vorgehen:

Erstellen Sie zunächst im Verzeichnis HOME ein Unterverzeichnis MPFIT (Multiple linear and Polynomial data FITting, Mehrfache lineare Anpassung und Polynomanpassung) und geben Sie das Unterverzeichnis MPFIT ein. Geben Sie im Unterverzeichnis dieses Programm ein:

```
« → X y « X TRAN X * INV X TRAN * y * » »
```

und speichern Sie es in einer Variablen MTREG (MulTiple REGression).

Geben Sie anschließend die Matrizen **X** und **b** in den Stack ein:

[[1,1.2,3.1,2][1,2.5,3.1,2.5][1,3.5,4.5,2.5][1,4,4.5,3][1,6,5,3.5]]

ENTER ENTER (speichern Sie eine zusätzliche Kopie)

[5.7,8.2,5.0,8.2,9.5] ENTER

Drücken Sie VAR MTREG. Das Ergebnis lautet: [-2.1649…,–0.7144…,-1.7850…,7.0941…], d. h.

$$y = -2.1649–0.7144 \cdot x_1 -1.7850\times10^{-2} \cdot x_2 + 7.0941 \cdot x_3.$$

Der Stack des Taschenrechners sollte den Wert von Matrix X und Vektor b enthalten. Die angepassten Werte von y werden durch **y** = **X**·**b** ermittelt. Drücken Sie daher einfach ×, um folgendes Ergebnis zu erhalten: [5.63.., 8.25.., 5.03.., 8.23.., 9.45..].

Vergleichen Sie diese angepassten Werte wie in der folgenden Tabelle mit den ursprünglichen Werten:

x_1	x_2	x_3	y	an y angepa sst

1.20	3.10	2.00	5.70	5.63
2.50	3.10	2.50	8.20	8.25
3.50	4.50	2.50	5.00	5.03
4.00	4.50	3.00	8.20	8.23
6.00	5.00	3.50	9.50	9.45

Polynomanpassung

Gegeben sei der x-y-Datensatz $\{(x_1,y_1), (x_2,y_2), \ldots, (x_n,y_n)\}$. Angenommen, wir möchten ein Polynom der Ordnung p an diesen Datensatz anpassen. Mit anderen Worten, wir möchten eine Datenanpassung der Form $y = b_0 + b_1 \cdot x + b_2 \cdot x^2 + b_3 \cdot x^3 + \ldots + b_p \cdot x^p$ durchführen. Sie können die Näherung der kleinsten Quadrate an die Werte der Koeffizienten $\mathbf{b} = [b_0 \;\; b_1 \;\; b_2 \;\; b_3 \ldots b_p]$ durch Erzeugen der Matrix **X** erhalten.

$$\begin{bmatrix} 1 & x_1 & x_1^2 & x_1^3 & \ldots & x_1^{p-1} & y_1^{p} \\ 1 & x_2 & x_2^2 & x_2^3 & \ldots & x_2^{p-1} & y_2^{p} \\ 1 & x_3 & x_3^2 & x_3^3 & \ldots & x_3^{p-1} & y_3^{p} \\ . & . & . & . & & . & . \\ . & . & . & . & . & . & . \\ 1 & x_n & x_n^2 & x_n^3 & \ldots & x_n^{p-1} & y_n^{p} \end{bmatrix}$$

Der Vektor der Koeffizienten wird dann mit $\mathbf{b} = (\mathbf{X}^T \cdot \mathbf{X})^{-1} \cdot \mathbf{X}^T \cdot \mathbf{y}$ ermittelt, wobei **y** den Vektor $\mathbf{y} = [y_1 \, y_2 \ldots y_n]^T$ darstellt.
In Kapitel 10 wurde die einem Vektor $\mathbf{x} = [x_1 \; x_2 \ldots x_m]$ entsprechende Vandermonde-Matrix definiert. Die Vandermonde-Matrix ist mit der Matrix **X** für die Polynomanpassung vergleichbar, enthält jedoch lediglich *n* und nicht (*p+1*) Spalten.
Wir können die Funktion VANDERMONDE zum Erstellen der Matrix **X** verwenden, wenn wir die folgenden Regeln beachten:

Wenn $p = n-1$, ist $\mathbf{X} = \mathbf{V}_n$.
Wenn $p < n-1$, entfernen Sie die Spalten $p+2, \ldots, n-1, n$ aus $\mathbf{V}_n$, um **X** zu erzeugen.

Wenn $p > n-1$, fügen Sie die Spalten $n+1, \ldots, p-1, p+1$ zu $\mathbf{V}_n$ hinzu, um die Matrix $\mathbf{X}$ zu erzeugen.

In Schritt 3 dieser Liste müssen wir berücksichtigen, dass die Spalte i ($i= n+1, n+2, \ldots, p+1$) den Vektor $[x_1^i \; x_2^i \; \ldots \; x_n^i]$ darstellt. Wenn wir für x eine Liste von Datenwerten und keinen Vektor verwenden, d. h. $\mathbf{x} = \{ x_1 \; x_2 \; \ldots \; x_n \}$, können wir die Folge $\{ x_1^i \; x_2^i \; \ldots \; x_n^i \}$ einfach berechnen. Wir wandeln anschließend diese Liste in einen Vektor um und verwenden das Menü COL, um diese Spalten der Matrix $\mathbf{V}_n$ hinzuzufügen, bis $\mathbf{X}$ fertig gestellt ist.

Nachdem **X** erstellt und der Vektor **y** verfügbar ist, entspricht die Berechnung des Koeffizientenvektors **b** der mehrfachen linearen Anpassung (der vorherigen Matrixanwendung). Somit können wir ein Programm zum Berechnen der Polynomanpassung schreiben, das auf dem bereits für die mehrfache lineare Anpassung verwendeten Programm aufbaut. Wir müssen diesem Programm die oben aufgeführten Schritte 1 bis 3 hinzufügen.

Der *Algorithmus* für dieses Programm kann daher wie folgt geschrieben werden:

Vektoren **x** und **y** derselben Dimension als Listen eingeben. (Hinweis: Da für die Funktion VANDERMONDE eine Liste als Eingabe verwendet wird, empfiehlt es sich, die (x,y)-Daten als Liste einzugeben.) Außerdem den Wert von p eingeben.

- n = Größe von Vektor **x** bestimmen
- Mit der Funktion VANDERMONDE die Vandermonde-Matrix $\mathbf{V}_n$ für die eingegebene Liste **x** generieren
- If $p = n-1$, then
  $\mathbf{X} = \mathbf{V}_n$,
  Else If $p < n-1$
  die Spalten $p+2, \ldots, n$ aus $\mathbf{V}_n$ entfernen, um **X** zu erstellen
  (FOR-Schleife und COL- verwenden)
  Else
  die Spalten $n+1, \ldots, p+1$ zu $\mathbf{V}_n$ hinzufügen, um **X** zu erstellen
  (FOR-Schleife, x^i berechnen, in Vektor konvertieren, COL+ verwenden)
- **y** in Vektor konvertieren

- **b** mit dem Programm MTREG berechnen (siehe obiges Beispiel für mehrfache lineare Anpassung)

Es folgt die *Übertragung des Algorithmus* in ein Programm in der Sprache USER RPL. (Weitere Informationen über das Programmieren erhalten Sie in Kapitel 21.)

«	Programm starten
→ x y p	Listen x und y sowie p (Ebenen 3,2,1) eingeben
«	Unterprogramm 1 starten
x SIZE → n	Größe der Liste x bestimmen
«	Unterprogramm 2 starten
x VANDERMONDE	x in Stack ablegen, $\mathbf{V}_n$ ermitteln
IF 'p<n-1' THEN	Durch diese IF-Klausel wird Schritt 3 des Algorithmus implementiert
n	n in Stack ablegen
p 2 +	p+1 berechnen
FOR j	Schleife j = n-1, n-2, …, p+1, step = -1 starten
j COL– DROP	Spalte entfernen und aus Stack löschen
-1 STEP	FOR-STEP-Schleife beenden
ELSE	
IF 'p>n-1' THEN	
n 1 +	n+1 berechnen
p 1 +	p+1 berechnen
FOR j	Schleife mit j = n, n+1, …, p+1 starten
x j ^	$\mathbf{x}^j$ als Liste berechnen
OBJ→ →ARRY	Liste in Feld konvertieren
j COL+	Spalte zu Matrix hinzufügen
NEXT	FOR-NEXT-Schleife beenden
END	Zweite IF-Klausel beenden
END	Erste IF-Klausel beenden Das Ergebnis ist **X**
y OBJ→ →ARRY	Liste **y** in ein Feld konvertieren
MTREG	Von Programm MTREG verwendetes **X** und **y**

→NUM	In Dezimalformat konvertieren
»	Unterprogramm 2 beenden
»	Unterprogramm 1 beenden
»	Hauptprogramm beenden

Speichern Sie das Programm in einer Variablen POLY (POLYnomanpassung).

Verwenden Sie als *Beispiel* die folgenden Daten, um eine Polynomanpassung mit p = 2, 3, 4, 5, 6 zu erhalten.

x	y
2.30	179.72
3.20	562.30
4.50	1969.11
1.65	65.87
9.32	31220.89
1.18	32.81
6.24	6731.48
3.45	737.41
9.89	39248.46
1.22	33.45

Da wir für die Anpassung von Polynomen unterschiedlicher Ordnung dieselben x-y-Daten verwenden, sollten Sie die Listen der Datenwerte x und y in den Variablen xx bzw. yy speichern. Auf diese Weise müssen wir die Daten nicht bei jeder Anwendung des Programms POLY erneut eingeben. Gehen Sie daher wie folgt vor:

{ 2.3 3.2 4.5 1.65 9.32 1.18 6.24 3.45 9.89 1.22 } ENTER 'xx' STO▸
{179.72 562.30 1969.11 65.87 31220.89 32.81 6731.48 737.41 39248.46 33.45} ENTER 'yy' STO▸

Verwenden Sie zum Anpassen der Daten an die Polynome folgende Eingabe:
xx yy 2 POLY. Ergebnis: [4527.73 -3958.52 742.23]
d. h. $y = 4527.73\text{-}39.58x+742.23x^2$
xx yy 3 POLY. Ergebnis: [–998.05 1303.21 -505.27 79.23]

d. h. $y = -998.05+1303.21x-505.27x^2+79.23x^3$

xx yy 4 POLY. Ergebnis: [20.92 –2.61 –1.52 6.05 3.51]

d. h. $= 20.92-2.61x-1.52x^2+6.05x^3+3.51x^4$.

xx yy 5 POLY. Ergebnis: [19.08 0.18 –2.94 6.36 3.48 0.00]

d. h. $y = 19.08+0.18x-2.94x^2+6.36x^3+3.48x^4+0.0011x^5$

xx yy 6 POLY. Ergebnis: [-16.73 67.17 –48.69 21.11 1.07 0.19 0.00]

d. h. $y = -16.73+67.17x-48.69x^2+21.11x^3+1.07x^4+0.19x^5+0.0058x^6$

Auswählen der besten Anpassung

Wie Sie aus den obigen Ergebnissen ersehen, können Sie ein beliebiges Polynom an einen Satz von Daten anpassen. Nun ergibt sich die Frage, welche Anpassung für die Daten am besten geeignet ist. Für die Auswahl der besten Anpassung können mehrere Kriterien verwendet werden:

- Der Korrelationskoeffizient r. Dieser Wert ist auf den Bereich $-1 < r < 1$ eingeschränkt. Je näher r an dem Wert +1 oder –1 ist, desto besser ist die Datenanpassung.
- Die Summe der quadratischen Fehler SSE. Hierbei handelt es sich um die Größe, die durch den Ansatz der kleinsten Quadrate minimiert werden soll.
- Ein Diagramm von Residuen. Hierbei handelt es sich um ein Diagramm der Fehler, die den einzelnen ursprünglichen Datenpunkten entsprechen. Wenn diese Fehler vollständig zufällig sind, darf das Residuendiagramm keinen bestimmten Verlauf aufweisen.

Bevor wir diese Kriterien in einem Programm implementieren, stellen wir folgende Definitionen vor:

Für die Vektoren **x** und **y** der an die Polynomgleichung anzupassenden Daten erstellen wir die Matrix **X** und berechnen mit ihr einen Vektor der Polynomkoeffizienten **b**. Wir können mit **y'** = **X**·**b** einen *Vektor angepasster Daten* **y**' berechnen.

Ein *Fehlervektor* wird mit e = y – **y**' berechnet.

Die *Summe der quadratischen Fehler* ist gleich dem Quadrat des Betrags des Fehlervektors, d. h. $SSE = |\mathbf{e}|^2 = \mathbf{e} \bullet \mathbf{e} = \Sigma\, e_i^2 = \Sigma\, (y_i - y'_i)^2$.

Zum Berechnen des Korrelationskoeffizienten müssen wir zunächst die *Gesamtquadratsumme* SST (Sum of Squared Totals) ermitteln, die als $SST = \Sigma\, (y_i - \bar{y})^2$ definiert ist, wobei $\bar{y}$ den *Mittelwert* der ursprünglichen Werte von y darstellt, d. h. $\bar{y} = (\Sigma y_i)/n$.

Für SSE und SST ist der Korrelationskoeffizient definiert durch

$$r = [1-(SSE/SST)]^{1/2}.$$

Im Folgenden wird das neue Programm mit der Berechnung von SSE und r dargestellt (wir empfehlen nochmals, auf der letzten Seite dieses Kapitels nachzulesen, wie die Variablen- und Befehlsnamen in dem Programm erstellt werden):

«	Programm starten
→ x y p	Listen x und y sowie Zahl p eingeben.
«	Unterprogramm 1 starten.
x SIZE → n	Größe der Liste x bestimmen
«	Unterprogramm 2 starten
x VANDERMONDE	x in Stack ablegen, $\mathbf{V}_n$ ermitteln
IF 'p<n-1' THEN	Diese IF-Klausel entspricht Schritt 3 des Algorithmus.
n	n in Stack ablegen.
p 2 +	p+1 berechnen.
FOR j	Schleife j = n-1 bis p+1, step = -1 starten
j COL– DROP	Spalte entfernen und aus Stack löschen.
-1 STEP	FOR-STEP-Schleife beenden.
ELSE	
IF 'p>n-1' THEN	
n 1 +	n+1 berechnen

p 1 +	p+1 berechnen
FOR j	Schleife mit j = n, n+1, …, p+1
starten	
x j ^	$\mathbf{x}^j$ als Liste berechnen
OBJ→ →ARRY	Liste in Feld konvertieren
j COL+	Spalte zu Matrix hinzufügen
NEXT	FOR-NEXT-Schleife beenden
END	Zweite IF-Klausel beenden
END	Erste IF-Klausel beenden Erzeugt **X**
y OBJ→ →ARRY	Liste **y** in ein Feld konvertieren
→ X yv	Matrix und Feld als X und y eingeben
«	Unterprogramm 3 starten
X yv MTREG	Von Programm MTREG verwendetes **X** und **y**
→NUM	Bei Bedarf Umwandlung in Fließkommawerte
→ b	Als b übergebener Ergebnisvektor
«	Unterprogramm 4 starten
b yv	**b** und yv in Stack ablegen
X b *	**X**·**b** berechnen
-	**e** = **y** - **X**·**b** berechnen
ABS SQ DUP	SSE berechnen, Kopie erstellen
y ΣLIST n /	$\bar{y}$ berechnen
n 1 →LIST SWAP CON	Vektor mit *n* Werten von $\bar{y}$ erstellen
yv – ABS SQ	SST berechnen
/	SSE/SST berechnen
NEG 1 + √	r = [1–SSE/SST]$^{1/2}$ berechnen
"r" →TAG	Tagergebnis als "r"
SWAP	Ebene 1 und 2 des Stacks vertauschen
"SSE" →TAG	Tagergebnis als SSE
»	Unterprogramm 4 beenden
»	Unterprogramm 3 beenden
»	Unterprogramm 2 beenden
»	Unterprogramm 1 beenden

⪢ Hauptprogramm beenden

Speichern Sie dieses Programm unter dem Namen POLYR, um auf die Berechnung des Korrelationskoeffizienten r hinzuweisen.

Wenn das Programm POLYR für Werte zwischen 2 und 6 verwendet wird, wird die folgende Tabelle mit Werten des Korrelationskoeffizienten r und der Summe der quadratischen Fehler SSE erzeugt:

p	*r*	*SSE*
2	0.9971908	10731140.01
3	0.9999768	88619.36
4	0.9999999	7.48
5	0.9999999	8.92
6	0.9999998	432.61

Während der Korrelationskoeffizient für alle Werte von p in der Tabelle sehr nahe an 1.0 ist, variieren die Werte von SSE stark. Der kleinste Wert von SSE entspricht p = 4. Sie können somit die bevorzugte Polynomanpassung für die ursprünglichen x-y-Daten wie folgt auswählen:

$$y = 20.92-2.61x-1.52x^2+6.05x^3+3.51x^4.$$

Kapitel 19
Zahlen in verschiedenen Basen

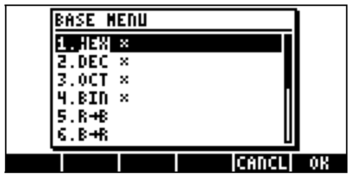
In diesem Kapitel zeigen wir Beispiele für Zahlenberechnungen in anderen Basen als der Dezimalbasis.

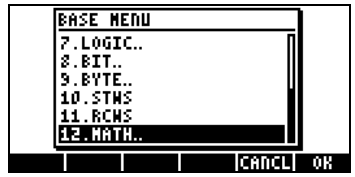
Definitionen

Das Zahlensystem, dass für das tägliche Rechnen verwendet wird, ist als *Dezimal*system bekannt,da es 10 (Latein, deca)Stellen,nämlich 0-9, verwenden, um eine wirkliche Zahl zu schreiben.Anderseits verwenden Computer ein System, das auf zwei möglichen Zuständen basiert oder dem *Binär*system. Diese beiden Zustände werden durch 0 und 1 dargestellt, AN und AUS oder Hochspannung und Niedrigspannung.Computer verwenden auch Zahlensysteme, die auf acht Stellen (0-7), also dem *Oktal*system, und sechzehn Stellen (0-9, A-F) oder *hexadezimal* basieren.Wie im Dezimalsystem bestimmt die relative Position der Stellen den Wert. Im Allgemeinen kann eine Zahl n in Basis b als Zahlenreihe n = $(a_1a_2 \ldots a_n.c_1c_2 \ldots c_m)_b$ geschrieben werden. Der "Punkt" teilt *n* "integere" Stellen von m "dezimalen" Stellen. Der Zahlenwert umgewandelt in unser übliches Dezimalsystem wird wir wie folgt berechnet: $n = a_1 \cdot bn^{-1} + a_2 \cdot b^{n-2} + \ldots + a_n b^0 + c_1 \cdot b^{-1} + c_2 \cdot b^{-2} + \ldots + c_m \cdot b^{-m}$. Zum Beispiel: $(15.234)_{10} = 1 \cdot 10^1 + 5 \cdot 10^0 + 2 \cdot 10^{-1} + 3 \cdot 10^{-2} + 4 \cdot 10^{-3}$, and $(101.111)_2 = 1 \cdot 2^2 + 0 \cdot 2^1 + 1 \cdot 2^0 + 1 \cdot 2^{-1} + 1 \cdot 2^{-2} + 1 \cdot 2^{-3}$

Das Menü BASE

Während der Rechner normalerweise mit dem Dezimalsystem bedient wird, können Sie auch Berechnungen mit dem Binär-,Oktal-oder Hexadezimalsystem durchführen.Viele der Funktionen zur Manipulierung der Zahlensysteme außer dem Dezimalsystem sind über das Menü BASE verfügbar und erreichbar über ⮣ BASE (die Taste 3). Wenn das Systemflag 117 auf die CHOOSE-Kästchen gesetzt ist, zeigt das Menü BASE die folgenden Einträge:





Ist die Systemmarkierung 117 auf die Menüs SOFT eingestellt, zeigt das Menü BASE das Folgende:

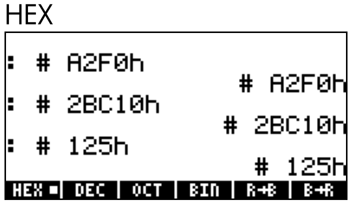
Mit diesem Format wird deutlich, dass die Einträge LOGIC, BIT und BYTE selbst im Menü BASE Untermenüs sind. Diese Menüs werden später in diesem Kapitel besprochen.

Die Funktionen HEX, DEC, OCT und BIN

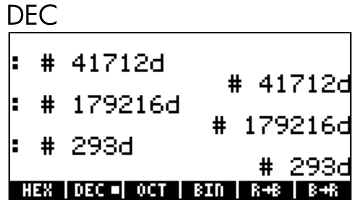
Zahlen in Nicht-Detimalsystemen wird ein #-Symbol im Rechner vorangestellt. Das Symbol # ist readily verfügbar als ← # (die Taste 3). Zur Auswahl welches Zahlensystem (derzeitige Basis) für Zahlen verwendet wird, denen ein # vorangestellt wird, wählen Sie eine der folgenden Funktionen im ersten Menü BASE, d.h., HEX(adezimal), DEC(dezimal), OCT(oktal), oder BIN(är). Wenn beispielsweise HEX▪ ausgewählt wird, wird jede Zahl, die in den Rechner geschrieben wird und mit einem # beginnt, als Hexadezimalzahl geschrieben. So können Sie Zahlen wie #53, #A5B, etc. in dieses System schreiben. Wenn andere systeme ausgwählt werden, werden die Zahlen automatisch in die neue aktuelle Basis umgewandelt.

Die folgenden Beispiele zeigen dieselben Zahlen mit vorangestelltem #-Symbol für verschiedene Basen:

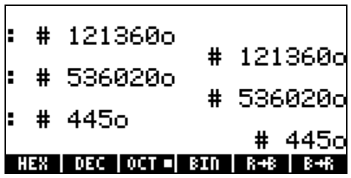
HEX



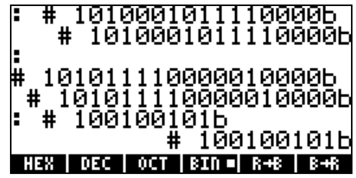
DEC



OCT



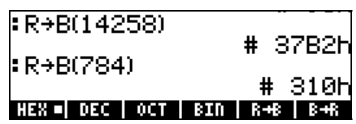
BIN



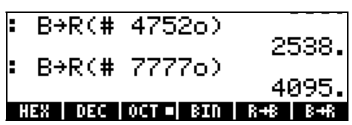
Da das Dezimalsystem (DEC) 10 Stellen hat (0,1,2,3,4,5,6,7,8,9), hat das Hexadezimalsystem (HEX) 16 Stellen (0,1,2,3,4,5,6,7,8,9,A,B,C,D,E,F), das Oktalsystem (OCT) hat 8 Stellen (0,1,2,3,4,5,6,7), und das Binärsystem (BIN) hat nur 2 Stellen (0,1).

Umwandlung zwischen Zahlensystemen

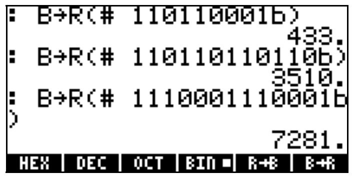
Egal welches Zahlensystem gewählt wurde,es wird als Binärsystem bezeichnet, um die Funktionen R→B and B→R verwenden zu können.Wenn beispielsweise HEX■ ausgewählt ist, wandelt die Funktion B→R jede Hexadezimalzahl (mit vorangestelltem #) in eine Dezimalzahl um, während die Funktion R→B in die entgegengesetzte Richtung umwandelt. Probieren Sie die folgenden Beispiele aus, HEX ist die derzeitige Basis:

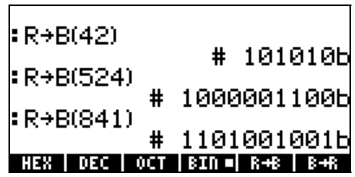


Die folgenden Beispiele zeigen Umwandlungen mit dem Oktalsystem als Basis:



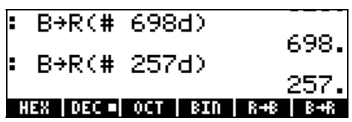
Wir zeigen auch Transformationen mit dem Binärsystem als aktuelle Basis:

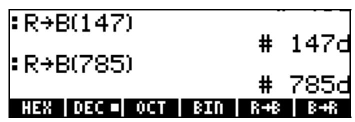




Wir weisen darauf hin, dass Sie immer, wenn Sie eine Nummer mit vorangestelltem # eingeben, als Eintrag die eingebenene Zahl mit vorangestelltem # und nachgestelltem h, o, oder b (hexadezimal, oktal oder binär) erhalten. Der als Suffix verwendete Buchstabe hängt davon ab, welches nichtdezimale Zahlensystem ausgewählt wurde, d.h. HEX, OCT oder BIN.

Probieren Sie die folgenden Umwandlungen, damit Sie sehen, was passiert, wenn Sie DEC auswählen:





Die einzige Auswirkung, die die Auswahl des DEC (Dezimalsystem) hat, sind die Dezimalzahlen mit vorangestelltem #-Symbol, die mit dem Suffix d versehen werden.

Wortgröße

Die Wortgröße ist die Bitanzahl in einem Binärprojekt.Standardmäßig beträgt die Wordgröße 64 Bites.Die Funktion RCWS (ReCall WordSize) zeigt die aktuelle Wortgröße.Die Funktion STWS (SeT the WordSize) ermöglicht dem Benutzer, die Wortgröße auf irgendeine Zahl zwischen 0 und 64 einzustellen.

Die Änderung der Wortgröße beeinflusst die Weise, wie integere Rechenoperationen ausgeführt. Wenn beispielsweise ein binäres Integer die aktuelle Wortgröße überschreitet, fallen die Bits vorne weg, bevor eine Rechenoperation mit einer solchen Zahl durchgeführt werden kann.

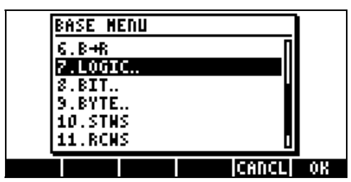
Rechenoperationen mit binären Integerzahlen

Die Rechenoperationen Addition, Subtraktion, Zeichenwechsel, Multiplikation und Division sind für binäre Integerzahlen definiert. Einige Beispiele werden für Addition und Subtraktion bei verschiedenen aktuellen Basen unten gezeigt:

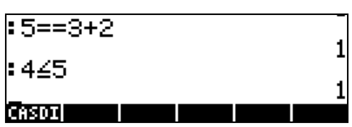
```
          #A02h + #12Ah = #B2Ch
        #2562d + #298d = #2860d
        #5002o + #452o = #5454o
#101000000010b + #100101010b = #101100101100b

          #A02h - #12Ah = #8D8h
        #2562d - #298d = #2264d
        #5002o - #452o = #4330o
#101000000010b - #100101010b = #100011011000b
```

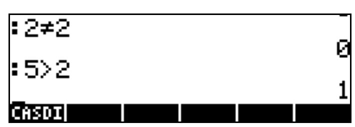
Das Menü LOGIC

Das Menü LOGIC ist über BASE (→ BASE) erreichbar und bietet die folgenden Funktionen:



Die Funktionen AND, OR, XOR (exklusives OR) und NOT sind logische Funktionen. Bei der Eingabe in diese Funktionen handelt es sich zwei Werte oder Ausdrücke (einer im Fall von NOT), die als binäre logische Ergebnisse ausgedrückt werden können, d.h. 0 oder 1. Zahlenvergleiche mit den Vergleichsoperatoren =, ≠, >, <, ≤ und ≥ sind logische Anweisungen, die entweder wahr (1) oder falsch (0) sein können. Einige Beispiele für logische Anweisungen werden unten gezeigt:



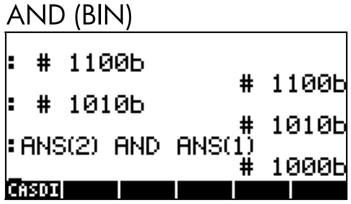


Die Funktionen AND, OR, XOR und NOT können auf die Vergleichs-Anweisungen mit den folgenden Regeln angewendet werden:

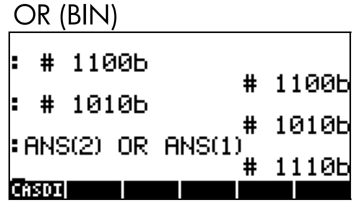
1 AND 1 = 1	1 AND 0 = 0	0 AND 1 = 0	0 AND 0 = 0
1 OR 1 = 1	1 OR 0 = 1	0 OR 1 = 1	0 OR 0 = 0
1 XOR 1 = 0	1 XOR 0 = 1	0 XOR 1 = 1	0 XOR 0 = 0
NOT(1) = 0	NOT(0) = 1		

Diese Funktionen können zur Bildung logischer Anweisungen zu Programmierzwecken verwendet werden. Im Kontext dieses Kapitels werden sie zur Darstellung der Bit-by-Bit-Ergebnisse entsprechend der oben gezeigten Regeln verwendet. In den nachfolgenden Beispielen, wird das Basis Zahlensystem in Klammern angegeben:

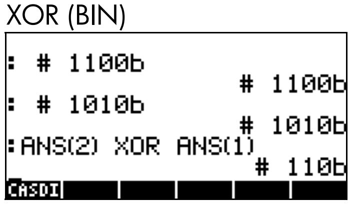
AND (BIN)



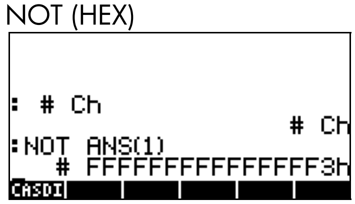
OR (BIN)



XOR (BIN)



NOT (HEX)



Das Menü BIT

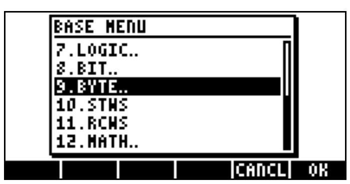
Das Menü BIT ist über BASE (→ BASE) erreichbar und bietet die folgenden Funktionen:

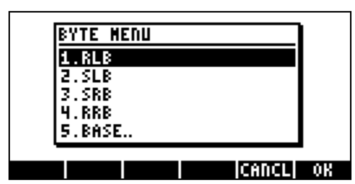
Die Funktionen RL, SL, ASR, SR, RR sind im Menü BIT enthalten und werden zur Änderung von Bits in einem binären Integer verwendet. Die Definition dieser Funktionen wird unten gezeigt:

RL: Rotate Left one bit (ein Bit links drehen), z.B., #1100b → #1001b
SL: Shift Left one bit (ein Bit nach links schieben), z.B., #1101b → #11010b
ASR: Arithmetic Shift Right one bit (ein Bit arithmetisch rechts drehen), z.B., #1100010b → #110001b
SR: Shift Right one bit (ein Bit nach rechts schieben), z.B., #11011b →#1101b
RR: Rotate Right one bit (ein Bit rechts drehen), z.B., #1101b → #1110b

Das Menü BYTE

Das Menü BIT ist über BASE (↱ BASE) erreichbar und bietet die folgenden Funktionen:





Die Funktionen RLB, SLB, SRB, RRB, sind im Menü BIT enthalten und werden zur Änderung von Bits in einem binären Integer verwendet. Die Definition dieser Funktionen wird unten gezeigt:

RLB: Rotate Left one byte (ein Byte links drehen), z.B., #1100b → #1001b
SLB: Shift Left one byte (ein Bit nach links schieben), z.B., #1101b → #11010b

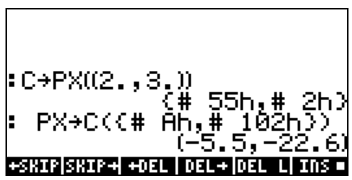
SRB: Shift Right one byte (ein Byte nach rechts schieben), z.B., #11011b →#1101b

RRB: Rotate Right one byte (ein Byte rechts drehen), z.B., #1101b →#1110b

Hexadezimalzahlen für Pixelreferenzen

Viele Druckoptionsspezifikationen verwenden Pixelreferenzen als Eingabe, z.B., { #332h #A23h } #Ah 0. 360. ARC, um einen Bogen aus einem Kreis zu zeichnen. Wir verwenden die Funktionen C→PX und PX→C, um schnell zwischen Benutzereinheitskoordinaten und Pixelreferenzen umzurechnen. Diese Funktionen finden Sie in der Befehlsreferenz (↱ CAT).

Unten werden einige Beispiele gezeigt:



Kapitel 20
Besonders anfertigen der Menüs und Tastatur

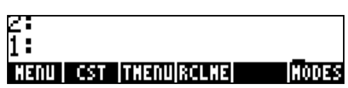
Durch den Gebrauch von den vielen Rechnermenüs sind Sie mit dem Betrieb der Menüs für eine Vielzahl von Anwendungen vertraut geworden. Auch sind Sie mit den vielen Funktionen vertraut, die vorhanden sind indem Sie die Schlüssel in der Tastatur, verwenden, ob durch ihre Hauptfunktion oder indem Sie sie kombinieren mit die rechte umschalttaste (↱), die linke umschalttaste (↰) oder die ALPHA (ALPHA). Schlüssel In diesem Kapitel stellen wir Beispiele der kundengebundenen Menüs und der Tasten zur Verfügung, daß Sie nützlich finden können in Ihren eigenen Anwendungen.

Besonders anfertigen der Menüs

Ein kundenspezifisches Menü ist ein Menü, das vom Benutzer gemacht wird. Die Spezifikationen für das Menü werden in den reservierten Variablen CST gespeichert. So um ein Menü zu machen müssen Sie diese Variable mit den Eigenschaften die Sie in Ihrem Menü anzeigen möchten zusammenfügen, mit den Tätigkeiten, die durch die soft Menüschlüssel angefordert werden. Um Beispiele des Besonders anfertigens der Menüs zuzeigen, müssen wir die Systemmarkierungsfahne 117 auf SOFT Menü einstellen. Tun Sie dies sicher, bevor Sie fortfahren (sehen Kapitel 2 für Anweisungen in der Einstellung der Systemmarkierungsfahnen).

Das PRG/MODES/MENU Menü

Die Befehle, die nützlich sind, wenn man Menüs besonders anfertigt, werden durch das MENU-Menü zur Verfügung gestellt, das zugänglich ist durch das PRG Menü (↰ PRG). Systemmarkierungsfahne 117 einstellend zum SOFT Menü, produziert das Reihenfolge ↰ PRG NXT MODES MENU das soft Menü des folgenden MENU:



Die vorhandenen Funktionen sind:

MENU : Aktiviert ein Menü, dessen Zahl gegeben wird

CST : Hinweis auf der CST Variable z.B., ⇁ CST zeigt CST Inhalt.
TMENU : Gebrauch anstelle vom MENU, um ein temporäres Menü außen zu machen ohne den Inhalt von CST zu überschreiben
RCLMENU : Bringt Menüzahl des gegenwärtigen Menüs zurück

Menü numeriert (RCL MENU- und MENU-Funktionen)

Jedes vorbestimmte Menü hat eine Zahl, die zu ihm angebracht wird. Z.B. nehmen Sie an, daß Sie aktivieren das MTH Menü (↰ MTH). Dann mit dem Funktion Katalog (⇁ CAT) finden Sie Funktion RCLMENU und aktivieren Sie sie. In der ALG Modus einfach Betätigen sie ENTER nach RCLMENU() im Schirm gezeigt wird. Das Resultat ist die Nr. 3,01. So können Sie das MTH Menü aktivieren, indem Sie MENU(3.01), in ALG oder 3.01 MENU, in RPN verwenden.

Die meisten Menüs können aktiviert werden, ohne ihre Zahlen zu kennen durch das Verwenden der Tastatur. Es gibt jedoch einige Menüs, die durch die Tastatur nicht zugänglich sind. Z.B. ist der soft Menü STATS nur zugänglich, indem er Funktion MENU verwendet. Seine Zahl ist 96,01. Benutzen Sie MENU(96.01) im ALG Modus oder 96.01 MENU im RPN Modus, um das soft Menü STAT zu erhalten.

Anmerkung: Die Nr. 96,01 in diesem Beispiel bedeutet das erste (01) Unterprogramm von Menü 96.

Kundenspezifische Menüs (MENU und TMENU-Funktionen)

Nehmen Sie an, daß Sie vier Funktionen für eine bestimmte Anwendung aktivieren müssen. Sagen Sie, das, das Sie in der Lage sein müssen, die Funktionen EXP, LN und GAMMA schnell zugänglich zu machen und ! (ALPHA ⇁ 2) und Sie möchten sie in ein soft Menü legen, daß Sie für eine Weile aktiv halten. Sie konnten dies tun, indem Sie ein temporäres Menü mit Funktion TMENU oder ein dauerhafteres, Menü mit Funktion MENU machen. Der Hauptunterschied ist, daß Funktion MENU variablen CST macht, während TMENU das nicht tut. Wenn variabler CST dauerhaft in Ihrem Unterverzeichnis gemacht ist, können Sie das Menü mit den Spezifikationen in CST immer reaktivieren, indem Sie ↰ CUSTOM betätigen. Mit TMENU sind die

Menüspezifikationen verloren, nachdem Sie das temporäre Menü durch ein anderes ersetzen.

Z.B. im RPN Modus, wird ein Menü gemacht, indem man verwendet:

{EXP LN GAMMA !} `ENTER` TMENU `ENTER`

oder

{EXP LN GAMMA !} `ENTER` MENU `ENTER`

um das folgende Menü zuproduzieren:

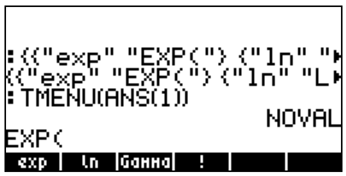
Um Irgendwelche jener Funktionen zu aktivieren, müssen Sie einfach das Funktion Argument (eine Zahl), eintragen und betätigen dann den entsprechenden soft Menüschlüssel.

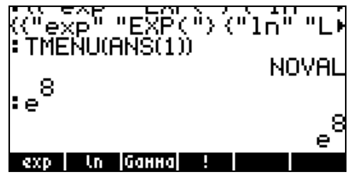
Im ALG Modus ist die einzufüren Liste als Argument der Funktion TMENU oder MENU schwieriger:

{{“exp”,”EXP(“},{“ln”,”LN(“},{“Gamma”,”GAMMA(“},{“!”,”!(“}}

Der Grund für dieses ist, daß, im RPN Modus, die Befehl Namen beide soft Menümarkierungen und -befehle sind. Im ALG Modus produzieren die Befehl Namen keine Tätigkeit, da ALG Funktionen von Klammern und von Argumenten gefolgt werden müssen. In der Liste oben gezeigt (für den ALG Modus), innerhalb jedes Sub-list haben Sie einen markierung für den Schlüssel z.B., “exp”, gefolgt von der Reihenfolge, daß die Funktion in den Stapel eingetragen wird, damit das Argument zur Funktion an der Aufforderung z.B.“EXP(“ geschrieben kan worden. Wir brauchen uns nicht um die schließenden Klammern zu sorgen, weil der Rechner die Klammern durchführt vor der Durchführung, der Funktion. Implementierung von Funktion TMENU im ALG Modus mit der Argumentliste, die oben gezeigt wird, ist wie folgt. Zuerst tragen wir die Liste ein, dann produzieren wir das temporäre Menü (sehen Sie Menüschlüsselmarkierung), indem wir Funktion

TMENU(ANS(1)) verwenden.   Wir zeigen auch, in der linken Seite, das Resultat des Betätigens des exp soft Menüschlüssel d.h. das Aufforderung EXP. Nach schreiben 8 ENTER wird das Resultat des Betriebes in der rechten Seite gezeigt:





Eine einfachere Version des Menüs kann definiert werden, indem man

MENU({{"EXP(","LN(","GAMMA(","!("}) verwendet.

Erhöhtes RPN Menü
Die Liste, die oben für den ALG Modus dargestellt wird, kann etwas geändert werden, um im RPN Modus zu verwenden.   Die geänderte Liste sieht wie dieses aus:

{{"exp",EXP},{"ln",LN},{"Gamma",GAMMA},{"!",!}}

Sie können diese, Liste mit TMENU oder MENU im RPN Modus zu verwenden versuchen, um zu überprüfen, daß Sie das gleiche Menü wie früher im ALG Modus erreicht erhalten.

Menüspezifikation und CST Variable

Von den zwei Übungen, die oben gezeigt werden, beachten wir daß die allgemeinste Menüspezifikation Liste eine Anzahl von Sub-lists umfassen, der in Zahl gleich sind an die in Ihrem kundenspezifischen Menü angezeigte Einzelteilen. Jeder Sub-list enthält einen markierung für den Menüschlüssel, der gefolgt wird von einer Funktion, Ausdruck, Markierung oder von anderem Gegenstand, den Effekt des Menüschlüssels ersetzt.   Obacht muß ausgeübt werden, wenn man die Menüliste im ALG Modus gegen RPN Modus spezifiziert. Im RPN Modus kann die Menüschlüsseltätigkeit einfach ein Rechnerbefehl (z.B., EXP, LN, etc., wie oben gezeigt) sein, während im ALG

Modus es eine Zeichenkette mit dem Befehlseingabeformat sein muß dessen Argument vom Benutzer zur Verfügung gestellt werden muß, bevor er ` betätigt ENTER und den Befehl durchführt. Die Beispiele oben veranschaulichen den Unterschied.

Die allgemeine Form von der Argumentliste für Befehle TMENU, oder MENU im ALG Modus ist
{"label1","function1(","ls1(","rs1("), {"label2", "function2(","ls2(","rs2("),…)

Während, im RPN Modus, die Argumentliste dieses Format hat

{"label1", function1, ls1, rs1}, {"label2", function2, ls2, rs2},…}

In diesen Spezifikationen stellen function1, function2 usw. die Hauptverwendung der Taste dar, während ls1, ls2, …., usw. die Verwendung der Taste in Kombination mit der linken Shift-Taste darstellen. Ähnlich stellen, rs1, rs2, …, usw. die Verwendung der Taste in Kombination mit der rechten Shift-Taste dar. Diese Liste wird in variablem CST gespeichert, wenn das Befehl MENU benutzt wird. Sie können eine andere CST Variable in jedem Unterverzeichnis haben, und Sie können den gegenwärtigen Inhalt von CST mit denen anderer Variablen immer ersetzen, welche die richtig formatierte Liste speichern, um ein anderes kundenspezifisches Menü zu produzieren.

Anmerkung: Sie können ein 21x8 GROB (siehe Kapitel 22) benutzen, um ein Symbol im Funktionstastenmenü zu erstellen. Als Übung versuchen Sie im RPN-Modus:

{{GROB 21 8 00000EF908FFF900FFF9B3FFF9A2FFF9A3FFF9A0FFF388FF "hp" }}
ENTER MENU

Damit wird das hp Logo auf der Taste F1 dargestellt. Drücken Sie F1 wird der Text 'hp' in der Befehlszeile eingegeben.

Besonders anfertigen der Tastatur

Jeder Schlüssel in der Tastatur kann durch zwei Zahlen gekennzeichnet werden, die ihre Reihe und Spalte darstellen. Z.B. ist der VAR Schlüssel (VAR) in Reihe 3 von Spalte 1 und wird gekennzeichnet als Schlüssel 31. Jetzt da jeder Schlüssel bis 10 Funktionen hat, die mit ihm verbunden sind, wird jede Funktion durch Dezimalziffern zwischen 0 und 10, entsprechend den folgenden Spezifikationen spezifiziert:

,0 oder 1, Taste allein	,01 oder ,11, nicht zutreffend
,2, Taste in Kombination mit ←	,21, Taste simultan mit ←
,3, Taste in Kombination mit →	,31, Taste simultan mit →
,4, Taste in Kombination mit ALPHA	,41, Taste simultan mit ALPHA
,5, Taste in Kombination mit ALPHA ←	,51, ALPHA Taste simultan mit ←
,6, Taste in Kombination mit ALPHA →	,61, ALPHA Taste simultan mit →

So wird die VAR Funktion als Schlüssel 31.0, oder 31.1 gekennzeichnet, während die UPDIR Funktion Schlüssel 31.2 ist, die copy Funktion Schlüssel 31.3, das Versalien j ist Schlüssel 31.4, und Klein j ist Schlüssel 31.5. (Schlüssel 31,6 ist nicht definiert). Im allgemeinen wird ein Schlüssel durch die Anordnung XY.Z beschrieben, in der X = Reihe Zahl, Y = Spalte Zahl, Z = verschiebung.

Wir können einen gegebenen Schlüssel mit dem USER-Schlüssel kombinieren (link-umschalttaste verbunden mit dem ALPHA Schlüssel oder ← USER), um eine kundengebundene Schlüsseltätigkeit zu verursachen. Prinzipiell kann die gesamte Tastatur neu definiert werden, um eine Anzahl von kundengebundenen Betrieben durchzuführen.

Das PRG/MODES/MENU Menü

Die Befehle, die nützlich sind, wenn man Menüs besonders anfertigt, werden durch das Tasten-Menü zur Verfügung gestellt, das zugänglich ist durch das PRG Menü (← PRG). Systemmarkierungsfahne 117 einstellend zum SOFT Menü, produziert das Reihenfolge ← PRG NXT MODES KEYS

produziert der folgenden KEY soft Menü:

Die vorhandenen Funktionen sind:

ASN : Weist einen Gegenstand einem Schlüssel zu, der von XY.Z angegeben wird
STOKEYS : Speichert verbraucherbestimmte Schlüsselliste
RCLKEYS : Zeigt verbraucherbestimmte Schlüsselliste
DELKEYS : entfernt einen oder mehr Schlüssel im gegenwärtigen verbraucherbestimmten Schlüsselliste, die Argumente sind irgendein 0, entfernt ganz verbraucherbestimmte Schlüssel oder XY.Z, entfernt Schlüssel-XY.Z.

Rückruf verbraucherbestimmte Schlüsselliste

Verwenden Sie Befehl RCL KEY um die gegenwärtige verbraucherbestimmte Schlüsselliste zu sehen. Vor allen verbraucherbestimmten Schlüsselanweisungen sollte das Resultat eine Liste sein, die den Buchstaben S enthält d.h., {S}.

Zuweisen einer Gegenstand zu ein verbraucherbestimmten Schlüssel

Nehmen Sie an, daß Sie Zugang zum altmodischen PLOT-Befehl haben möchten, der zuerst mit dem HP 48G Reihe Rechner eingeführt wird, aber z.Z. nicht direkt vorhanden ist von der Tastatur. Die Menüzahl für dieses Menü ist 81.01. Sie können dieses Menü aktiv sehen, indem Sie verwenden :

ALG Modus: MENU(81.01)
RPN Modus: 81.01 ENTER MENU ENTER

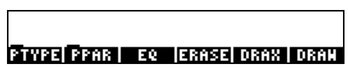
Wenn Sie eine schnelle Weise haben möchten, dieses Menü von der Tastatur zu aktivieren, konnten Sie dieses Menü dem GRAPH -Schlüssel (F3) zuweisen dessen Tastenkennzeigen 13,0 ist d.h. erste Reihe, dritte Spalte, Hauptfunktion. Um einen Gegenstand einer Schlüssel zu zuweisen, gebrauchfunktion ASN, wie folgt:

ALG Modus: `ASN(<<MENU(81.01)>>,13.0)`
RPN Modus: `<< 18.01 MENU >>` [ENTER] `13.0` [ENTER] `ASN`

Ein anderes nützliches Menü ist die Vorlage SOLVE das Menü (beschrieben am Ende von Kapitel 6 in diesem Führer), das durch das Verwenden von [→](hold) [7] aktiviert werden kann.

Gebrauch verbraucherbestimmte Schlüssel

Um diesen verbraucherbestimmten Schlüssel laufen zu lassen, betätigen Sie [←]USER bevor Sie den [F3] Schlüssel betätigen. Beachten Sie, daß nach betätigen von [←]USER der Schirm die Spezifikation `1USR` in der zweiten Anzeige Linie zeigt. Betätigend von [←]USER [F3] für dieses Beispiel, sollten Sie das PLOT-Menü zurückgewinnen, wie folgt:



Wenn Sie mehr als einen verbraucherbestimmten Schlüssel haben und mehr als ein von ihnen hintereinander laufen lassen möchten, können Sie die Tastatur im USER-Modus verriegeln, indem Sie [←]USER [←]USER eintragen, bevor Sie die verbraucherbestimmten Schlüssel betätigen. Wenn die Tastatur im USER-Modus verriegelt ist, wird die Spezifikation `USR` in der zweiten Anzeige Linie gezeigt. Um das Tastatur zu entriegeln betätigen [←]USER noch einmal.

Entfernen einen verbraucherbestimmten Schlüssel

Um die zuweizung oben durch geführt zu entfernen, Gebrauch die funktion DELKEYS, wie folgt:

ALG Modus: `DELKEYS(13.0)`

RPN mode: `13.0` [ENTER] `DELKEYS` [ENTER]

Zuweisen mehrfachen verbraucherbestimmten Schlüssel

Die einfachste Weise, einige verbraucherbestimmte zuzuweisen ist eine Liste von Befehlen und Schlüsselspezifikationen zur Verfügung zustellen. Z.B. nehmen Sie an, daß wir die drei trigonometrischen Funktionen (SIN, COS, TAN) und die drei Hyperbelfunktionen (SINH, COSH, TANH) Schlüsseln F1 bis F6 beziehungsweise als verbraucherbestimmte Schlüssel zuweisen. Im RPN Modus gebrauch:

{SIN,11.0,COS,12.0,TAN,13.0,SINH,14.0,COSH,15.0,TANH,16.0} ENTER STOKEYS ENTER

Im ALG Modus gebrauch:

STOKEYS({"SIN(' ,11.0, "COS(", 12.0, "TAN(", 13.0, "SINH(", 14.0, "COSH(", 15.0, "TANH(", 16.0}) ENTER

Lassen Sie diese Schlüssel laufen, indem Sie z.B. im RPN Modus verwenden:

5 ← USER F1   4 ← USER F2   6 ← USER F3
2 ← USER F4   1 ← USER F5   2 ← USER F6

Entfernen Sie allen verbraucherbestimmten Schlüsselgebrauch mit:

ALG Modus: DELKEYS(0)   RPN mode: 0 DELKEYS

Prüfen Sie, ob die Benutzer-Schlüssel Definitionen durch das Verwenden der Funktion RCLKEYS entfernt wurden.

Kapitel 21
Programmieren mit RPL

RPL ist eine Programmiersprache, die allgemein zur Programmierung von Rechner verwendet wird. Programmteile können im Zeileneditor zusammengeführt werden, indem sie in entsprechender Reihenfolgen zwischen die Programm-Container « » eingefügt werden. Da die meisten Anwender über eine größere Erfahrung in der Programmierung im RPN-Modus besitzen, werden die meisten Beispiele in diesem Kapitel im RPN-Modus dargestellt. Außerdem empfehlen wir, um die Eingabe der Befehle zu vereinfachen, dass Sie das System-Flag 117 auf SOFT-Menüs einstellen. Die Programme arbeiten natürlich auch im ALG-Modus, wenn im RPN-Modus getestet und Fehler beseitigt wurden. Wenn Sie es vorziehen, im ALG-Modus zu arbeiten, lernen Sie einfach, wie in RPN programmiert wird, und setzen dann den Rechner auf den ALG-Modus zurück, um mit den Programmen zu arbeiten. Auf der letzten Seite in diesem Kapitel finden Sie ein einfaches Beispiel der RPL-Programmierung im ALG-Modus.

Programmierbeispiel

In den vorangegangenen Kapiteln dieses Handbuches haben wir Ihnen bereits einige Programme vorgestellt, die für verschiedene Applikationen verwendet werden können (beispielsweise wurden Im Kapitel 20 CRMC und CRMT vorgestellt, die dazu dienen, eine Matrix anhand einer Anzahl von Listen zu erstellen). In diesem Absatz zeigen wir Ihnen ein einfaches Programm, um Konzepte zur Programmierung des Rechners vorzustellen. Das Programm, das wir erstellen werden, dient zur Definition der Funktion $f(x) = \sinh(x)/(1+x^2)$ und akzeptiert als Argument Listen (x kann z.B., wie in Kapitel 8 beschrieben, eine Liste von Zahlen sein). In Kapitel 8 wurde angeführt, dass das Plus-Zeichen (+) als Verkettungsoperator für Listen dient und nicht Glied für Glied addiert. Um die Listen Glied für Glied zu addieren müssen Sie den ADD-Operator verwenden. Um also die oben genannte Funktion zu definieren, verwenden wir das folgende Programm:

```
«'x' STO x SINH 1 x SQ ADD / 'x' PURGE »
```

Um das Programm einzugeben, folgen Sie den nachstehenden Anweisungen:

Tastenfolge:	Erzeugt:	Interpretiert als:
[→] « »	«	Start des RPL-Programms
['] [ALPHA] [←] [X] [▶] [STO▶]	'x' STO	Speichert Ebene 1 in der Variablen x
[ALPHA] [←] [X]	x	Setzt x in Ebene 1
[←] MTH HYP SINH	SINH	Berechnet sinh von Ebene 1
[1] [SPC] [ALPHA] [←] [X] [←] x^2	1 x SQ	1 eingeben und x^2 berechnen
[←] MTH MTH LIST ADD	ADD	Berechnen $(1+x^2)$,
[÷]	/	dann dividieren
['] [ALPHA] [←] [X] [▶]	'x'	
[←] PRG MEM DIR PURGE	PURGE	Variable x löschen
[ENTER]		Programm in Ebene 1

Um das Programm zu speichern, verwenden Sie diese Tastenfolge:
['] [ALPHA] [←] [G] [STO▶]

Drücken Sie [VAR], um das Variablen-Menü wieder aufzurufen und untersuchen Sie g(3.5), indem Sie den Wert des Arguments in Ebene 1 eingeben ([3] [•] [5] [ENTER]) und dann [g] drücken. Das Ergebnis ist 1.2485…, d.h., g(3.5) = 1.2485. Ermitteln Sie auch g({1 2 3}), indem Sie die Liste in Ebene 1 der Anzeige eingeben: [←] {} [1] [SPC] [2] [SPC] [3] [ENTER] und [g] drücken. Wurde CAS auf den EXACT Modus eingestellt, ist das Ergebnis {SINH(1)/2 SINH(2)/5 SINH(3)/10}. Wurde CAS auf den APPROXIMATE Modus eingestellt, ist das Ergebnis {0.5876.. 0.7253… 1.0017…}.

Globale und lokale Variablen und Unterprogramme

Das Programm [g], wie oben definiert, kann angezeigt werden

```
« 'x' STO x SINH 1 x SQ ADD / 'x' PURGE »
```

indem Sie [→] [g] eingeben.

Beachten Sie, dass das Programm den Variablen-Namen x verwendet, um den Wert der Ebene 1 des Stacks für die Programmschritte 'x' STO zu speichern. Die Variable x wird, während das Programm ausgeführt wird, wie

alle Variablen, die Sie vorher gespeichert haben, in ihrem Variablen-Menü gespeichert. Nach der Berechnung der Funktion löscht das Programm die Variable x, so dass sie nach Beendigung des Programms nicht mehr im Variablen-Menü angezeigt wird. Würde die Variable x nach Beendigung des Programms nicht gelöscht, stünde sie auch weiterhin zur Verfügung. Aus diesem Grunde bezeichnet man die Variable x, wie sie in diesem Programm verwendet wird, als *eine globale Variable*. Eine Auswirkung bei der Verwendung von x als globale Variable liegt darin, dass eine vorher definierte Variable mit dem Namen x von der Variablen x, die das Programm gerade benutzt überschrieben und nach der Programmausführung aus dem Variablen-Menü entfernt wird.
Aus Programmierersicht ist eine globale Variable eine Variable, auf die der Anwender nach der Programmausführung zurückgreifen kann. Es ist möglich, eine lokale Variable innerhalb eines Programms zu definieren, die dann nur im Programm gilt und auf die nach Programmausführung nicht mehr zugegriffen werden kann. Das vorhergehende Programm könnte hierzu folgendermaßen modifiziert werden:

```
« → x « x SINH 1 x SQ ADD / » »
```

Das Pfeilsymbol (→) erhält man aus der Kombination der rechten Shift-Taste ↱ mit der Taste 0, d.h., ↱ →. Beachten Sie auch, dass es ein weiteres Paar Programmsymbole (« ») gibt, die anzeigen, dass innerhalb des Hauptprogramms ein Unterprogramm existiert, hier « `x SINH 1 x SQ ADD /` ». Das Hauptprogramm startet mit der Kombination → `x`. Hierdurch wird der Wert in Ebene 1 des Stacks der lokalen Variablen _x zugeordnet. Danach fährt das Programm innerhalb des Unterprogramms fort, indem es x in den Stack schreibt und *SINH(x)* ermittelt, danach 1 in den Stack schreibt, x in den Stack schreibt, x quadriert und *1* zu x addiert. Zum Schluss wird der Stack der Ebene 2 (*SINH(x)*) durch den Stack der Ebene 1 ($1+x^2$) dividiert. Danach geht die Programmsteuerung wieder an das Hauptprogramm über. Da sich aber zwischen dem ersten Satz der Symbole (»), die das Programm abschließen, keine weiteren Befehle mehr befinden, wird das Programm beendet. Der letzte Wert im Stack, hier *SINH(x)/* ($1+x^2$), wird als Ergebnis des Programms ausgegeben.

Die Variable x des letzten Programms belegt nie einen Platz unter den Variablen Ihres Variablen-Menüs. Sie wird innerhalb des Speichers des Rechners verarbeitet, ohne dass sie Einfluss auf gleichlautende Variablen Ihres Variablen-Menüs hat. Aus diesem Grunde bezeichnet man die Variable x, wie sie hier lokal innerhalb eines Programms verwendet wird, als *eine lokale Variable*.

Anmerkung: Wollen Sie das Programm modifizieren, setzen Sie den Programmnamen in den Stack (ENTER) und verwenden dann . Benutzen Sie dann die Pfeiltasten ( ), um sich in dem Programm zu bewegen. Verwenden Sie die Zurück/Löschtaste, , um nicht gewünschte Zeichen zu löschen. Um Programm-Container (d.h., « ») hinzuzufügen, verwenden Sie «». Da die Symbole immer paarweise erscheinen, müssen Sie sie am Anfang und am Ende eines Unterprogramms aufrufen und dann jeweils eine Komponente mit der Löschtaste löschen, um das erforderliche Programm zu erzeugen, hier: « → x « x SINH 1 x SQ ADD / » ». Drücken Sie nach dem Editieren des Programms ENTER . Das modifizierte Programm wird wieder in die Variable gespeichert.

Geltungsbereich Globale Variablen

Jede Variable, die Sie im HOME-Verzeichnis, in anderen Verzeichnissen oder Unterverzeichnissen definieren, werden aus Programmsicht als *globale Variable* betrachtet. Der Geltungsbereich einer solchen Variablen, d.h., *die Position im Verzeichnisbaum, wo auf die Variable zugegriffen werden kann*, hängt von der Position der Variablen in diesem Baum ab (siehe Kapitel 2).

Die Regel zur Feststellung des Geltungsbereiches einer Variablen ist wie folgt: Eine Variable ist zugänglich vom Verzeichnis in dem sie sich befindet oder von jedem Unterverzeichnis dieses Verzeichnisses, wenn sich in dem entsprechenden Unterverzeichnis nicht eine Variable mit gleichem Namen befindet. Daraus ergeben sich folgende Konsequenzen:

- Eine globale Variable im HOME-Verzeichnis ist von jedem Verzeichnis, das zum HOME-Verzeichnis gehört, zugänglich, falls sie nicht in einem Verzeichnis oder Unterverzeichnis neu definiert wurde.
- Wenn Sie eine Variable innerhalb eines Verzeichnisses oder Unterverzeichnisses neu definieren, hat diese Definition Vorrang vor allen anderen Definitionen innerhalb eines übergeordneten Verzeichnisses.
- Wird ein Programm gestartet, das auf eine gegebene globale Variable zugreift, verwendet das Programm den Wert der globalen Variablen aus dem Verzeichnis, aus dem das Programm gestartet wurde. Befindet sich im Startverzeichnis des Programms keine Variable mit entsprechendem Namen, sucht das Programm alle übergeordneten Verzeichnisse bis einschließlich des HOME-Verzeichnisses ab und verwendet dann den Wert der entsprechenden Variablen, die sich in einem Verzeichnis befindet, das dem Verzeichnis des Programms am nächsten liegt.

Auf ein in einem gegebenen Verzeichnis definiertes Programm kann von diesem und von jedem Unterverzeichnis dieses Verzeichnisses zugegriffen werden.

Alle diese Regeln scheinen für einen neuen Anwender zunächst verwirrend zu sein. Sie können aber alle auf die folgende Vereinfachung zurückgeführt werden: Erstellen Sie Verzeichnisse und Unterverzeichnisse mit sinnvollen Namen, um Ihre Daten zu organisieren. Stellen Sie sicher, dass sich alle globalen Variablen, die Sie benötigen, in dem entsprechenden Unterverzeichnis befinden.

Geltungsbereich Lokale Variablen

Lokale Variablen sind nur in einem Programm oder einem Unterprogramm aktiv. Daher ist Ihr Geltungsbereich nur auf das Programm oder Unterprogramm, in dem sie definiert sind, beschränkt. Ein Beispiel einer lokalen Variablen ist der Index einer FOR-Schleife (weiter unten in diesem Kapitel beschrieben), beispielsweise `« → n x « 1 n FOR j x NEXT n →LIST » »`

Das Menü PRG

In diesem Kapitel zeigen wir Ihnen das PRG-(Programmierungs)-Menü mit auf Funktionstasten geschalteter System-Flag 117. Mit dieser Einstellung werden die Untermenüs und Befehle im PRG-Menü als Funktionstasten dargestellt Dies erleichtert Ihnen bei der Programmerstellung die Eingabe der Befehle in den Zeileneditor.

Um das PRG-Menü aufzurufen, verwendet Sie die Tastenfolge ← PRG . Innerhalb des PRG-Menüs finden wir folgende Untermenüs (drücken Sie NXT um zu dem nächsten Untermenü zu wechseln):

Nachfolgend finden Sie eine kurze Beschreibung der Untermenüs und ihrer Untermenüs:

STACK: Funktionen zur Manipulation von Elementen im RPN-Stack

MEM: Funktionen zur Speichermanipulation
- DIR: Funktionen zur Manipulation von Verzeichnissen
- ARITH: Funktionen zur Manipulation von Indizes, die in Variablen gespeichert sind

BRCH: Kollektion von Untermenüs für Programmverzweigungen und Programmschleifen
- IF: IF-THEN-ELSE-END Befehl zur Programmverzweigung
- CASE: CASE-THEN-END Befehl zur Programmverzweigung
- START: START-NEXT-STEP Befehl zur Programmverzweigung
- FOR: FOR-NEXT-STEP Befehl für Schleifen
- DO: DO-UNTIL-END Befehl für Schleifen
- WHILE: WHILE-REPEAT-END Befehl für Schleifen

TEST: Vergleich von Operatoren, logischen Operatoren, Funktionen zum Flag-Test

TYPE: Funktionen zur Konvertierung von Objekttypen, zur Aufspaltung von Objekten usw.

LIST: Funktionen zur Manipulation von Listen

ELEM: Funktionen zur Manipulation von Elementen einer Liste
PROC: Funktionen um Verfahren Listen zuzuordnen

GROB: Funktionen zur Manipulation grafischer Objekte
PICT: Funktionen zur Zeichnung von Bilder auf der Grafikanzeige
CHARS: Funktionen zur Manipulation von Zeichenketten
MODES: Funktionen zur Änderung der Rechenmodi

FMT: Änderung des Zahlen- und Kommaformats
ANGLE: Änderung des Winkelmaßes und Koordinatensystems
FLAG: Setzen und Löschen von Flags, Prüfen des Flag-Status
KEYS: Definition und Aktivierung anwenderdefinierter Tasten (Kapitel 20)
MENU: Definition und Aktivierung anwenderdefinierter Menüs (Kapitel 20)
MISC: Verschiedene Modi ändern (Warnton, Uhr usw.)

IN: Funktionen für Programmeingaben
OUT: Funktionen für Programmausgaben
TIME: Zeitgesteuerte Funktionen

ALRM: Warnmanipulationen

ERROR: Funktionen zur Fehlerbehandlung

IFERR: IFERR-THEN-ELSE-END Befehl zur Fehlerbehandlung

RUN: Funktionen zum Programmstart und Fehlersuche

Navigation durch die RPN-Untermenüs

Starten Sie mit der Tastenfolge ← PRG . Drücken Sie dann die entsprechende Funktionstaste (z.B. MEM). Wollen Sie dann auf ein Untermenü des Menüs zugreifen (z.B. DIR innerhalb des Menüs MEM), drücken Sie die entsprechende Taste. Um in der Hierarchie ein Menü nach oben zu wechseln, drücken Sie NXT , bis Sie zu dem übergeordneten Menü (z.B. zu MEM vom Menü DIR aus) oder zum PRG-Menü (PRG) gelangen.

Funktionen der Untermenüs

Nachfolgend finden Sie eine Auflistung der Untermenüs des PRG-Menüs und ihrer jeweiligen Funktionen.

STACK	MEM/DIR	BRCH/IF	BRCH/WHILE	TYPE
DUP	PURGE	IF	WHILE	OBJ→

SWAP	RCL	THEN	REPEAT	→ARRY
DROP	STO	ELSE	END	→LIST
OVER	PATH	END		→STR
ROT	CRDIR		**TEST**	→TAG
UNROT	PGDIR	**BRCH/CASE**	==	→UNIT
ROLL	VARS	CASE	≠	C→R
ROLLD	TVARS	THEN	<	R→C
PICK	ORDER	END	>	NUM
UNPICK			≤	CHR
PICK3	**MEM/ARITH**	**BRCH/START**	≥	DTAG
DEPTH	STO+	START	AND	EQ→
DUP2	STO-	NEXT	OR	TYPE
DUPN	STOx	STEP	XOR	VTYPE
DROP2	STO/		NOT	
DROPN	INCR	**BRCH/FOR**	SAME	**LIST**
DUPDU	DECR	FOR	TYPE	OBJ→
NIP	SINV	NEXT	SF	→LIST
NDUPN	SNEG	STEP	CF	SUB
	SCONJ		FS?	REPL
MEM		**BRCH/DO**	FC?	
PURGE	**BRCH**	DO	FS?C	
MEM	IFT	UNTIL	FC?C	
BYTES	IFTE	END	LININ	
NEWOB				
ARCHI				
RESTO				

LIST/ELEM	**GROB**	**CHARS**	**MODES/FLAG**	**MODES/MISC**
GET	→GROB	SUB	SF	BEEP
GETI	BLANK	REPL	CF	CLK
PUT	GOR	POS	FS?	SYM
PUTI	GXOR	SIZE	FC?	STK
SIZE	SUB	NUM	FS?C	ARG
POS	REPL	CHR	FS?C	CMD

HEAD	→LCD	OBJ→	FC?C	INFO
TAIL	LCD→	→STR	STOF	
	SIZE	HEAD	RCLF	**IN**
LIST/PROC	ANIMATE	TAIL	RESET	INFORM
DOLIST		SREPL		NOVAL
DOSUB	**PICT**		**MODES/KEYS**	CHOOSE
NSUB	PICT	**MODES/FMT**	ASN	INPUT
ENDSUB	PDIM	STD	STOKEYS	KEY
STREAM	LINE	FIX	RECLKEYS	WAIT
REVLIST	TLINE	SCI	DELKEYS	PROMPT
SORT	BOX	ENG		
SEQ	ARC	FM,	**MODES/MENU**	**OUT**
	PIXON	ML	MENU	PVIEW
	PIXOF		CST	TEXT
	PIX?	**MODES/ANGLE**	TMENU	CLLCD
	PVIEW	DEG	RCLMENU	DISP
	PX→C	RAD		FREEZE
	C→PX	GRAD		MSGBOX
		RECT		BEEP
		CYLIN		
		SPHERE		

TIME	**ERROR**	**RUN**
DATE	DOERR	DBUG
→DATE	ERRN	SST
TIME	ERRM	SST↓
→TIME	ERR0	NEXT
TICKS	LASTARG	HALT
		KILL
TIME/ALRM	**ERROR/IFERR**	OFF
ACK	IFERR	
ACKALARM	THEN	
STOALARM	ELSE	
RCLALARM	END	

DELALARM
FINDALARM

Kürzel des PRG-Menüs

Viele der oben für das PRG-Menü aufgeführten Funktionen stehen auch auf andere Weise zur Verfügung:

- Vergleichsoperatoren (≠, ≤, <, ≥, >) sind über die Tastatur verfügbar.
- Viele Funktionen und Einstellungen im Untermenü MODES können über die Eingabefunktionen der MODE Taste aktiviert werden.
- Auf Funktionen des Untermenüs TIME kann über die Tastenfolge → TIME zugegriffen werden.
- Die Funktionen STO und RCL (im Untermenü MEM/DIR) sind über die Tasten STO▸ und ← RCL verfügbar.
- Die Funktionen RCL und PURGE (im Untermenü MEM/DIR) sind über das TOOL-Menü (TOOL) verfügbar.
- Wenn Sie im Untermenü BRCH die linke (←) oder die rechte Shift-Taste (→) drücken, ehe Sie eine Taste des Untermenüs drücken, werden Befehle des ausgewählten Untermenüs erstellt. Die funktioniert nur dann, wenn sich der Rechner im RPN-Modus befindet. Nachfolgend einige Beispiele:

← IF

← CASE

→ IF

→ CASE

← START

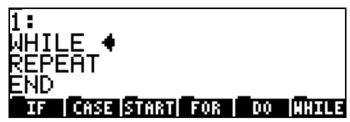
← FOR

↱ START

↱ FOR

↰ DO

↰ WHILE



Beachten Sie, das das Einfügezeichen (◆) nach dem Schlüsselwort des Befehls steht, so dass Sie direkt mit der Eingabe an der richtigen Stelle beginnen können.

Tastenfolge für häufig verwendete Befehle

Im Folgenden finden Sie Tastenfolgen, mit denen Sie häufig vorkommende Befehle zur numerischen Programmierung im PRG-Menü aufrufen können. Die Befehle werden nach den Menüs aufgeführt:

STACK

DUP	↰ PRG STACK DUP
SWAP	↰ PRG STACK SWAP
DROP	↰ PRG STACK DROP

MEM DIR

PURGE	↰ PRG MEM DIR PURGE
ORDER	↰ PRG MEM DIR ORDER

BRCH IF

IF	← PRG BRCH IF IF
THEN	← PRG BRCH IF THEN
ELSE	← PRG BRCH IF ELSE
END	← PRG BRCH IF END

BRCH CASE

CASE	← PRG BRCH CASE CASE
THEN	← PRG BRCH CASE THEN
END	← PRG BRCH CASE END

BRCH START

START	← PRG BRCH START START
NEXT	← PRG BRCH START NEXT
STEP	← PRG BRCH START STEP

BRCH FOR

FOR	← PRG BRCH FOR FOR
NEXT	← PRG BRCH FOR NEXT
STEP	← PRG BRCH FOR STEP

BRCH DO

DO	← PRG BRCH DO DO
UNTIL	← PRG BRCH DO UNTIL
END	← PRG BRCH DO END

BRCH WHILE

WHILE	← PRG BRCH WHILE WHILE
REPEAT	← PRG BRCH WHILE REPEA
END	← PRG BRCH WHILE END

TEST

==	← PRG TEST ≠
AND	← PRG TEST NXT AND
OR	← PRG TEST NXT OR
XOR	← PRG TEST NXT XOR
NOT	← PRG TEST NXT NOT
SAME	← PRG TEST NXT SAME

	SF	← PRG TEST NXT NXT SF
	CF	← PRG TEST NXT NXT CF
	FS?	← PRG TEST NXT NXT FS?
	FC?	← PRG TEST NXT NXT FC?
	FS?C	← PRG TEST NXT NXT FS?C
	FC?C	← PRG TEST NXT NXT FC?C
TYPE		
	OBJ→	← PRG TYPE OBJ→
	→ARRY	← PRG TYPE →ARRY
	→LIST	← PRG TYPE →LIST
	→STR	← PRG TYPE →STR
	→TAG	← PRG TYPE →TAG
	NUM	← PRG TYPE NXT NUM
	CHR	← PRG TYPE NXT CHR
	TYPE	← PRG TYPE NXT TYPE
LIST ELEM		
	GET	← PRG LIST ELEM GET
	GETI	← PRG LIST ELEM GETI
	PUT	← PRG LIST ELEM PUT
	PUTI	← PRG LIST ELEM PUTI
	SIZE	← PRG LIST ELEM SIZE
	HEAD	← PRG LIST ELEM NXT HEAD
	TAIL	← PRG LIST ELEM NXT TAIL
LIST PROC		
	REVLIST	← PRG LIST PROC REVLI
	SORT	← PRG LIST PROC NXT SORT
	SEQ	← PRG LIST PROC NXT SEQ
MODES ANGL		
	DEG	← PRG NXT MODES ANGL DEG
	RAD	← PRG NXT MODES ANGL DEG
MODES MENU		
	CST	← PRG NXT MODES MENU CST

	MENU	← PRG NXT MODES MENU MENU
	BEEP	← PRG NXT MODES MISC BEEP
IN		
	INFORM	← PRG NXT IN INFOR
	INPUT	← PRG NXT IN INPUT
	MSGBOX	← PRG NXT OUT MSGBO
	PVIEW	← PRG NXT OUT PVIEW
RUN		
	DBUG	← PRG NXT NXT RUN DBG
	SST	← PRG NXT NXT RUN SST
	SST↓	← PRG NXT NXT RUN SST↓
	HALT	← PRG NXT NXT RUN HALT
	KILL	← PRG NXT NXT RUN KILL

Programme zur Generierung von Zahlenlisten

Beachten Sie, dass die Funktionen des PRG-Menüs nicht die einzigen Funktionen sind, die sich zur Programmierung verwenden lassen. Eigentlich können fast alle Funktionen des Rechners in ein Programm integriert werden. So können Sie beispielsweise Funktionen des MTH-Menüs verwenden. Speziell können Sie Befehle für Listen, wie SORT, ΣLIST usw., aus dem Menü MTH/LIST verwenden.

Als zusätzliche Programmierübung mit den oben aufgeführten Tastenfolgen zeigen wir Ihnen drei Programme zur Erstellung oder Manipulation von Listen. Die Programmnamen und Listen sehen wie folgt aus:

LISC:

```
« → n x « 1 n FOR j x NEXT n →LIST » »
```

CRLST:

```
« → st en df « st en FOR  j j df STEP en st - df / FLOOR 1 +
→LIST » »
```

CLIST:

```
« REVLIST DUP DUP SIZE 'n' STO ΣLIST SWAP TAIL DUP SIZE 1 - 1
SWAP FOR j DUP ΣLIST SWAP TAIL NEXT 1 GET n →LIST REVLIST 'n'
PURGE »
```

Die Programme arbeiten wie folgt:

(1) *LISC*: Erzeugt eine Liste mit n Elementen, die alle gleich der Konstante c sind.
Ablauf: n eingeben, c eingeben, LISC drücken
Beispiel: 5 ENTER 6.5 ENTER LISC erzeugt folgende Liste: {6.5 6.5 6.5 6.5 6.5}

(2) *CRLST*: Erzeugt eine Liste mit Zahlen von n_1 bis n_2 mit einem Wertzuwachs Δn, d.h.,{n_1, n_1+Δn, n1+2·Δn, … n_1+N·Δn }, wobei N = floor((n_2-n_1)/Δn)+1.*Ablauf*: n_1 eingeben, n_2 eingeben, Δn eingeben, CRLST drücken
Beispiel: .5 ENTER 3.5 ENTER .5 ENTER CRLST erzeugt: {0.5 1 1.5 2 2.5 3 3.5}

(3) *CLIST*: Erzeugt eine Liste mit der kumulierten Summe der Elemente, d.h., wenn die Originalliste {x_1 x_2 x_3 … x_N} enthält, erzeugt CLIST die Liste:

$$\{x_1, x_1 + x_2, x_1 + x_2 + x_3, ..., \sum_{i=1}^{N} x_i\}$$

Ablauf: Legen Sie die Originalliste in Ebene 1 ab, drücken Sie CLIST.
Beispiel: {1 2 3 4 5} ENTER CLIST erzeugt {1 3 6 10 15}.

Beispiele zur sequentiellen Programmierung

Allgemein versteht man unter einem Programm jede Sequenz von Rechnerbefehle, die von den Programm-Containern « und » eingeschlossen wird. In einem Programm können Unterprogramme als Teil integriert sein Die Beispiele die vorher in diesem Handbuch aufgeführt wurden (z.B. in Kapitel 3 und 8), können in zwei Typen unterteilt werden: (a) Programme, die durch Definition einer Funktion erzeugt wurden und (b) Programme die eine Abfolge von Stack-Operationen ausführen. Diese beiden Arten von Programmen werden im Anschluss beschrieben. Die allgemeine Form dieser Programme kann man mit Eingabe→Verarbeitung→Ausgabe beschreiben. Daher werden diese Programme als sequentielle Programme bezeichnet.

Durch die Definition einer Funktion generierte Programme

Hierbei handelt es sich um Programme, die mit Hilfe der Funktion DEFINE (⬅ DEF) erstellt wurden, mit einem Argument in der Form:

'function_name(x_1, x_2, …) = Ausdruck mit den Variablen x_1, x_2, …'

Das Programm wird in der Variablen mit dem Namen `function_name` gespeichert. Wird das Programm mit der Tastenfolge ➡ function_name wieder in den Stack geladen, zeigt sich das Programm wie folgt:

« → x_1, x_2, … Ausdruck mit Variablen x_1, x_2, …'».

Um Variablen x_1, x_2, … im RPN-Modus zu untersuchen, geben Sie einige Variablen in richtiger Reihenfolge (d.h., x_1 zuerst, dann x_2, x_3, usw.) ein und drücken dann die Funktionstaste function_name. Der Rechner gibt das Ergebnis der Funktion function_name(x_1, x_2, ...) aus.

Beispiel: *Manning-Gleichung für weite rechteckige Kanäle.*
Als Beispiel verwenden Sie die Manning-Gleichung, die Durchfluss (Durchfluss je Weiteneinheit) q in einem weiten, rechteckigen, offenen Kanal berechnet:

$$q = \frac{C_u}{n} y_0^{5/3} \sqrt{S_0}$$

C_u ist eine Konstante, die von den verwendeten Maßeinheiten abhängt [C_u = 1.0 gilt für das International System (S.I.) und C_u = 1.486 gilt für das englische Maßsystem (E.S.)], n ist der Manning-Koeffizient für Rauhigkeit, der von der Kanaloberfläche und anderen Faktoren abhängt, y_0 ist die Flusstiefe und S_0 ist das dimensionslose Gefälle des Kanals.

Anmerkung: Werte des Manning-Koeffizienten n können Tabellen entnommen werde. Sie liegen im Allgemeinen zwischen 0,001 und 0,5.

Auch der Wert von C_u ist dimensionslos. Der Wert für y_0 muss jedoch die korrekte Dimension aufweisen, d.h., m für S.I. und Fuß für E.S. Das Ergebnis für q wird dann in der richten Maßeinheit ausgegeben, d.h., m^2/s in S.I. und ft^2/s in E.S. Die Manning-Gleichung ist daher nicht *Maßeinheits-konsistent*.

Gehen wir davon aus, dass wir die Funktion $q(C_u, n, y_0, S_0)$ erstellen wollen, um den Durchfluss q zu berechnen. Verwenden Sie den Ausdruck:

'q(Cu,n,y0,S0)=Cu/n*y0^(5/3)*√S0',

als Argument der Funktion DEFINE. Beachten Sie, dass der Exponent 5./3. in der Gleichung das Verhältnis von reellen Zahlen durch Dezimalkomma darstellt. Falls erforderlich drücken Sie [VAR], um die Variablenliste zu laden. An dieser Stelle befindet sich unter den Funktionstasten eine Variable mit der Bezeichnung [q]. Um den Inhalt von q anzuzeigen drücken Sie [→][q]. Das Programm, das durch die Definition der Funktion `q(Cu,n,y0,S0)` erstellt wurde, wird wie folgt angezeigt:

« → Cu n y0 S0 'Cu/n*y0^(5/3)*√S0' ».

Dies wird interpretiert als "geben Sie Cu, n, Y0, S0 nacheinander ein und berechne dann den Ausdruck". Um beispielsweise q für Cu = 1.0, n = 0.012, y0 = 2 m und S0 = 0.0001 im RPN-Modus zu berechnen, geben Sie ein:

1 [ENTER] 0.012 [ENTER] 2 [ENTER] 0.0001 [ENTER] [q]

Das Ergebnis ist 2.6456684 (oder q = 2.6456684 m^2/s).

Anstelle von [ENTER] können Sie die einzelnen Eingabedaten auch durch Leerzeichen trennen, wenn Sie sie in eine einzelne Stack-Zeile eingeben.

Programme zur Simulation eine Sequenz von Stack-Operationen

In diesem Fall wird davon ausgegangen, dass sich die Glieder, die zu der Operationssequenz gehören, im Stack befinden. Das Programm wird eingegeben, indem zunächst ein Programm-Container durch [→] [« »] geöffnet wird. Danach werden die einzelnen Operationen eingegeben. Drücken Sie, nachdem Sie alle Operationen eingegeben haben, ENTER , um das Programm abzuschließen. Wolle Sie das Programm nur ein einziges Mal verwenden, können Sie an dieser Stelle EVAL drücken, um das Programm mit den verfügbaren Eingabedaten durchzuführen. Soll das Programm immer wieder verwendet werden, muss es unter einem Variablennamen gespeichert werden.

Der beste Weg ein solches Programm zu beschreiben, ist anhand eines Beispiels:

Beispiel: Geschwindigkeitshöhe für einen rechteckigen Kanal
Wir wollen die Geschwindigkeitshöhe h_v in einem rechteckigen Kanal der Breite b mit einer Flusstiefe y und einer Durchflussmenge Q berechnen. Die spezifische Energie wird berechnet als $h_v = Q^2/(2g(by)^2)$, wobei g der Erdbeschleunigung ($g = 9.806\ m/s^2$ in S.I.-Einheiten oder $g = 32.2\ ft/s^2$ in E.S.-Einheiten) entspricht. Wollen wir h_v für Q = 23 cfs (Kubikfuß pro Sekunde = ft^3/s), b = 3 ft und y = 2 ft berechnen, geben wir ein: $h_v = 23^2/(2\cdot 32.2\cdot (3\cdot 2)^2)$. Im RPN-Modus rechnen wir interaktiv:

2 ENTER 3 × ← x^2 3 2 • 2 ×
2 × 2 3 ← x^2 ▶ ÷

Das Ergebnis ist 0.228174 oder h_v = 0.228174.

Um diese Berechnung als Programm aufzubauen, müssen sich die Eingabedaten (Q, g, b, y) in der Reihenfolge, in der sie bei der Berechnung verwendet werden, im Stack befinden. In Hinblick auf die Variablen Q, g, b und y wird die Kalkulation wie folgt eingegeben (bitte nicht eintippen):

y ENTER b × ← x^2 g × 2 × Q ← x^2 ▶ ÷

Wie Sie sehen steht y zuerst, dann kommen nacheinander b, g und Q. Daher benötigen wir die Variablen für diese Berechnung in umgekehrter Reihenfolge, d.h. (bitte nicht eintippen):

Q ENTER g ENTER b ENTER y ENTER

Für die vorgegebenen Werte verwenden wird:

23 ENTER 32.2 ENTER 3 ENTER 2 ENTER

Das Programm enthält nur die Tastenfolgen (oder Befehle), die sich daraus ergeben, dass die Eingabewerte der vorherigen interaktiven Kalkulation entfernt werden (bitte nicht eintippen):

y ENTER b × ← x^2 g × 2 × Q ← x^2 ▶ ÷

und nur noch die nachfolgend gezeigten Operationen behalten werden (bitte nicht eintippen):

ENTER × ← × 2 × ← x^2 ▶ ÷

Anmerkung: Verwenden Sie, wenn Sie das Programm eingeben, nicht die Taste ▶, sondern verwenden Sie stattdessen die Tastenfolge: ← PRG STACK SWAP.

Anders als bei der interaktiven Berechnung müssen wir in dem Programm die Stack-Ebenen 1 und 2 austauschen. Um das Programm zu schreiben, verwenden Sie die nachstehenden Tastenfolgen:

→ « »	Öffnen der Programmsymbole
×	Multiplikation von y mit b
← x^2	Quadrieren von (b·y)
×	$(b·y)^2$ g-mal multiplizieren
2 ×	Eingabe von 2 und Multiplikation mit g· $(b·y)^2$
← PRG STACK SWAP	Austausch von Q mit 2·g· $(b·y)^2$
← x^2	Quadrieren von Q
← PRG STACK SWAP	Austausch von 2·g· $(b·y)^2$ mit Q^2
÷	Division von Q^2 durch 2·g· $(b·y)^2$
ENTER	Programm abschließen

Das Programm sieht dann so aus:

```
« * SQ * 2 * SWAP SQ SWAP / »
```

Anmerkung: SQ ist die Funktion, die sich aus der Tastenfolge ← x^2 ergibt.

Wir wollen eine Kopie des Programms erstellen und diese unter dem Variablen-Namen hv speichern:

' ALPHA ← H ALPHA ← V STO▶

Im Funktionstastenmenü sollte jetzt eine neue Variable hv verfügbar sein. (Drücken Sie VAR, um die Variablenliste aufzurufen.) Das Programm, das sich noch im Stack befindet, kann mit der Funktion EVAL untersucht werden. Das Ergebnis sollte wie zuvor 0,228174 betragen. Das Programm ist für spätere Verwendung unter dem Namen hv verfügbar. Verwenden Sie beispielsweise für Q = 0.5 m^3/s, g = 9.806 m/s^2, b = 1.5 m und y = 0.5 m:

0.5 SPC 1.5 SPC 9.806 SPC 1.5 SPC 0.5 hv

Anmerkung: SPC wird hier als Alternative zu ENTER für die Dateneingabe verwendet.

Das Resultat ist nun 2.26618623518E-2, d.h., hv = $2.26618623518 \times 10^{-2}$ m.

Anmerkung: Da die programmierte Gleichung in hv dimensionskonsistent ist, können wir bei den Eingaben auch Einheiten verwenden.

Wie bereits früher erwähnt, handelt es sich bei den beiden Programmen in diesem Absatz um *sequentielle Programme*. Das bedeutet, das Programm folgt einem einzelnen Pfad, EINGABE→ BERECHNUNG →AUSGABE. Eine Verzweigung des Programms ist mit den Befehlen ← PRG BRCH . des Menüs möglich. Weitere Details hierzu finden Sie weiter unten.

Interaktive Eingabe in Programme

Bei den vorausgegangenen Programmbeispielen ist es für den Anwender nicht immer klar, in welcher Reihenfolge die Variablen vor der

Programmausführung im Stack angeordnet sein müssen. Bei dem Programm ▒▒, geschrieben als

« →Cu n y0 S0 'Cu/n*y0^(5/3)*√S0' »,

ist es immer möglich, die Programmdefinitionen neu in den Stack (↱ ▒▒) zu laden, um sehen zu können, in welcher Reihenfolge die Variablen eingegeben werden müssen, hier →Cu n y0 S0. Bei dem Programm ▒▒ bietet die Definition

« * SQ * 2 * SWAP SQ SWAP / »

keinen Hinweis, in welcher Reihenfolge die Daten eingegeben werden müssen. Es sei denn, Sie sind mit der RPN und der RPN-Programmiersprache sehr vertraut.

Ein Weg, das Resultat eines Programms als eine Formel zu überprüfen liegt darin, symbolische Variablen anstelle von numerischen Ergebnissen in den Stack einzugeben und dann das Programm mit diesen Werten arbeiten zu lassen. Hierzu muss das CAS (Calculator Algebraic System) des Rechners auf `symbolic` und `exact` eingestellt sein. Dies geschieht mit Hilfe von MODE CAS. Vergewissern Sie sich, das die Optionen `_Numeric` und `_Approx` deaktiviert sind.Drücken Sie OK OK, um zur normalen Anzeige zurückzukehren. Drücken Sie VAR, um das Variablen-Menü anzuzeigen.

Wir verwenden diese Methode, um festzustellen, welche Formel hinter der Verwendung des Programms ▒▒ verborgen ist, wie folgt: Wir wissen, dass für das Programm 4 Eingaben erforderlich sind. Daher verwenden wir die symbolischen Variablen S4, S3, S2 und S1:

ALPHA S 4 ENTER ALPHA S 3 ENTER ALPHA S 2 ENTER ALPHA S 1 ENTER

Drücken Sie als nächstes ▒▒. Die resultierende Formel sieht in etwa so aus:

'SQ(S4)/(S3*SQ(S2*S1)*2)',

wenn Ihre Anzeige nicht auf "Textbook"-Stil eingestellt ist, oder wie

$$\frac{SQ(S4)}{S3 \cdot SQ(S2 \cdot S1) \cdot 2}$$

wenn der “Textbook”-Stil gewählt wurde. Da wir wissen, dass die Funktion SQ( ) für x^2 steht, interpretieren wird das letzte Resultat als

$$\frac{S4^2}{2 \cdot S3 \cdot (S2 \cdot S1)^2},$$

welches die Positionen der verschiedenen Eingabeebenen des Stacks in der Formel anzeigt. Wenn wir dieses Resultat mit der von uns programmierten Originalformel vergleichen, d.h.,

$$h_v = \frac{Q^2}{2g(by)^2},$$

erkennen wir, dass wir y in Stack-Ebene 1 (S1), b in Stack-Ebene 2 (S2), g in Stack-Ebene 3 (S3) und Q in Stack-Ebene 4 (S4) eingeben müssen.

Prompt mit einer Eingabestring

Diese beiden Methoden sind nicht sehr effizient, um die Reihenfolge der Eingabedaten zu bestimmen. Sie können so aber zumindest den Anwender unterstützen, indem im die Namen der Variablen, die er einzugeben hat, angezeigt werden. Unter den verschiedenen Methoden, die die RPL-Sprache zur Verfügung stellt, ist die einfachste die Verwendung von Zeichenketten und der Funktion INPUT (⬅ PRG NXT IN INPUT), um die Eingabedaten zu laden.

Das folgende Programm fordert den Anwender auf, einen Wert für die Variable a einzugeben und legt diesen dann in die Stack-Ebene 1 ab.

```
« “Enter a:“ {“↵:a:“ {2 0} V } INPUT OBJ→ »
```

Das Programm enthält die Symbole :: (tag) und ↵(return), verfügbar über die Tastenfolgen ⬅ :: und ➡ ↵, zusammen mit der Taste •. Das Tag-

Symbol (::) dient dazu, Zeichenketten für die Ein- und Ausgabe zu markieren. Das Return-Symbol (↵) entspricht dem Drücken der Eingabetaste auf einem Computer. Die Zeichenketten zwischen den Anführungszeichen (“ “) werden direkt über die alphanumerische Tastatur eingegeben.

Speichern Sie das Programm in eine Variable mit der Bezeichnung INPTa (für INPuT a).
Starten Sie das Programm, indem Sie die Funktionstaste INPTa drücken.

Als Resultat wird der Anwender aufgefordert einen Wert für a einzugeben. Der Cursor wird direkt rechts von dem Prompt :a: gesetzt. Geben Sie beispielsweise 35 ein und drücken ENTER. Als Resultat erscheint der Eingabestring `:a:35` in der Stack-Ebene 1.

Funktion mit Eingabestring

Wenn Sie den oben erwähnten Code verwenden wollen, um beispielsweise die Funktion f(a) = 2*a^2+3 zu berechnen, können Sie das Programm folgendermaßen modifizieren:

```
« “Enter a: “ {“↵:a: “ {2 0} V }
INPUT OBJ→ → a « ‘2*a^2+3‘ » »
```

Speichern Sie dieses neue Programm unter dem Namen ‘FUNCa’ (FUNCtion von a):

Starten Sie das Programm, indem Sie FUNCa drücken. Sobald Sie dazu aufgefordert werden, geben Sie z.B. 2 ein und drücken ENTER. Als Resultat ergibt sich der algebraische Begriff $2a^2+3$. Dies ist aber falsch. Der Rechner bietet Ihnen Funktionen, mit deren Hilfe Sie Ihr Programm untersuchen und logische Fehler während der Programmausführung finden können

Fehlersuche im Programm

Um herauszufinden, was nicht funktioniert hat, verwenden wir die DEBUG-Funktion wie folgt:

['] [FUNCa] [ENTER]	Kopiert den Programmnamen in die Stack-Ebene 1
[←] PRG [NXT] [NXT] [RUN] [DBG]	Startet die Fehlersuche (den Debugger)
[SST↓]	Fehlersuche Schritt für Schritt, Ergebnis: "Enter a:"
[SST↓]	Ergebnis: {" ↵ a:" {2 0} V}
[SST↓]	Ergebnis: Anwender wird aufgefordert, für a einen Wert einzugeben
[2] [ENTER]	Geben Sie für a 2 ein. Ergebnis: "↵:a:2"
[SST↓]	Ergebnis: a:2
[SST↓]	Ergebnis: Stack leeren, ausführen von →a
[SST↓]	Ergebnis: Stack leeren, ins Unterprogramm springen «
[SST↓]	Ergebnis: '2*a^2+3'
[SST↓]	Ergebnis: '2*a^2+3', Unterprogramm verlassen »
[SST↓]	Ergebnis: '2*a^2+3', Hauptprogramm verlassen »

Weiteres Drücken von [SST↓] führt zu keinem weiteren Ergebnis, da wir das ganze Programm Schritt für Schritt durchlaufen sind.Diese Fehlersuche brachte keinerlei Anhaltspunkte, warum das Programm nicht den Wert von $2a^2+3$ für a = 2 berechnet. Um zu sehen, welchen Wert a im Unterprogramm hat, müssen wird den Debugger neu starten und den Wert von a im Unterprogramm untersuchen. Versuchen Sie folgendes:

[VAR]	Laden des Variablenmenüs
['] [FUNCa] [ENTER]	Kopiert den Programmnamen in die Stack-Ebene 1
[←] PRG [NXT] [NXT] [RUN] [DBG]	Startet die Fehlersuche (den Debugger)
[SST↓]	Fehlersuche Schritt für Schritt, Ergebnis:

	“Enter a:”
SST↓	Ergebnis: {“ ↵a:” {2 0} V}
SST↓	Ergebnis: Anwender wird aufgefordert, für a einen Wert einzugeben
2 ENTER	Geben Sie für a 2 ein. Ergebnis: “↵:a:2”
SST↓	Ergebnis: a:2
SST↓	Ergebnis: Stack leeren, ausführen von →a
SST↓	Ergebnis: Stack leeren, ins Unterprogramm springen «

An dieser Stelle befinden wir uns im Unterprogramm « ‘2*a^2+3’ », das die lokale Variable a benutzt. Um den Wert von a anzuzeigen, verwenden Sie:

ALPHA ← A EVAL	Tatsächlich wird für die lokale Variable a = 2 angezeigt.

Brechen wir hier den Debugger ab, da wir das Ergebnis ja bereits kennen. Um den Debugger abzubrechen, drücken Sie KILL. Es erscheint die Meldung <!> Interrupted zur Bestätigung des abgebrochenen Prozesses. Drücken Sie ON, um zur Normalanzeige des Rechners zurückzukehren.

Anmerkung: Jedes Mal, wenn Sie im Debugg-Modus SST↓ drücken, erscheint oben links in der Anzeige der auszuführende Programmschritt. Eine Funktionstaste SST ist auch als Teil des PRG-Menüs im Untermenü RUN verfügbar. Diese Funktion kann dazu verwendet werden, sofort ein Unterprogramm, das von einem Hauptprogramm aufgerufen wird, auszuführen. Beispiele zur Anwendung von SST folgen später.

Programm korrigieren

Die einzig mögliche Erklärung dafür, dass das Programm kein numerisches Ergebnis ausgibt, scheint auf das Fehlen des Befehls →NUM nach dem algebraischen Ausdruck ‘2*a^2+3’ zurückzuführen zu sein. Wir editieren das Programm und fügen die fehlende EVAL-Funktion ein. Das bearbeitete Programm sieht dann wie folgt aus:

```
« "Enter a: " {"↵a: " {2 0} V } INPUT
   OBJ→ → a « '2*a^2+3' →NUM » »
```

Speichern Sie das Programm wieder unter FUNCa und starten Sie es mit a = 2. Dieses Mal ist das Ergebnis 11, d.h., $2*2^2+3 = 11$.

Eingabestring für zwei oder drei Eingabewerte

In diesem Absatz werden wir innerhalb des HOME-Verzeichnisses ein Unterverzeichnis erstellen, das Beispiele für Eingabestrings mit ein, zwei oder drei Werten enthält.Dies sind allgemeine Eingabestrings, die in zukünftige Programme integriert werden können, unter Berücksichtigung, dass sich die Variablennamen entsprechend den Anforderungen der einzelnen Programme ändern können.

Erstellen wir als erstes ein Unterverzeichnis mit der Bezeichnung PTRICKS (Programmier-TRICKS), in das wir nützliche Programmteile ablegen, die wir in späteren komplexeren Programmen verwenden können. Um das Verzeichnis zu erstellen, wechseln Sie zunächst in das HOME-Verzeichnis. Verwenden Sie dann im HOME-Verzeichnis die nachstehende Tastenfolge, um das Unterverzeichnis PTRICKS anzulegen:

['] [ALPHA] [ALPHA] [P] [T] [R] [I] [C] [K] [S] [ENTER]	Geben Sie den Verzeichnisnamen 'PTRICKS' ein
[←] PRG MEM DIR CRDIR	Verzeichnis erstellen
[VAR]	Variablenliste laden

Ein Programm kann mehr als 3 Eingabewerte benötigen. Wenn wir Eingabestrings verwenden, wollen wir die Anzahl der Werte auf 5 beschränken, da wir üblicherweise nur 7 Stack-Ebenen sehen können. Wenn wir die Stack-Ebene 7 dazu benutzen, um für den Eingabestring einen Namen zu vergeben und die Ebene 6 frei halten, um die Anzeige einfach lesen zu können, bleiben zur Definition von Eingabevariablen nur noch die Stack-Ebenen 1 bis 5 übrig.

Programm mit Eingabestring für zwei Eingabewerte

Ein Programm für zwei Eingabewerte, beispielsweise a und b, sieht wir folgt aus:

```
« "Enter a and b: " {"↵:a:↵:b: " {2 0} V } INPUT OBJ→ »
```

Dieses Programm kann leicht erstellt werden, indem wir den Inhalt von INPTa modifizieren. Speichern Sie dieses Programm in die Variable INPT2.

Anwendung: Untersuchung einer Funktion mit zwei Variablen

Betrachten Sie das Gesetz über Ideale Gase pV = nRT, wo p = Gasdruck (Pa), V = Gasvolumen(m^3), n = Mol (gmol), R = universelle Gaskonstante = 8.31451_J/(gmol*K) und T = absolute Temperatur (K) sind.

Wir können den Druck p als Funktion von zwei Variablen V und T als p(V,T) = nRT/V für eine gegebene Gasmasse definieren, da n konstant bleibt. Gehen wir davon aus, das n = 0,2 gmol ist, ergibt sich als Funktion, die wir programmieren wollen,

$$p(V,T) = 8.31451 \cdot 0.2 \cdot \frac{T}{V} = (1.662902\_\frac{J}{K}) \cdot \frac{T}{V}$$

Wir definieren die Funktion, indem wir das folgende Programm eingeben

```
« →V T '(1.662902_J/K)*(T/V)' »
```

und es in die Variable `p` speichern.

Als nächstes muss der Eingabestring hinzugefügt werden, der den Anwender auffordert, die Werte für V und T einzugeben. Hierzu modifizieren Sie das Programm in `p` wie folgt:

```
« "Enter V and T: " {"↵ :V:↵ :T: " {2 0} V }
 INPUT OBJ→ → V T '(1.662902_J/K)*(T/V)' »
```

Speichern Sie das modifizierte Programm wieder unter . Drücken Sie to , um das Programm zu starten. Geben Sie für V = 0.01_m^3 und für T = 300_K ein und drücken dann ENTER. Als Ergebnis erscheint 49887.06_J/m^3. Die Einheit J/m^3 entspricht der Einheit Pascals (Pa), die im S.I.-System bevorzugte Maßeinheit für Druck.

> **Anmerkung**: Da wir absichtlich bei der Definition der Funktion Dimensionen verwendet haben, müssen auch die Werte mit Dimensionen eingegeben werden, um ein richtiges Resultat zu erzielen.

Programm mit Eingabestring für drei Eingabewerte

Ein Programm für drei Eingabewerte, beispielsweise a, b und c, sieht wir folgt aus:

```
« "Enter a, b and c:" {"↵ :a:↵ :b:↵ :c:" {2 0} V }
                INPUT  OBJ→ »
```

Dieses Programm kann leicht erstellt werden, indem wir den Inhalt von INPT2 wie oben dargestellt modifizieren. Speichern Sie das Programm unter der Bezeichnung INPT3. Mit diesem Programm beenden wir die Sammlung von Programmen, die es dem Anwender ermöglichen, einen, zwei oder drei Werte einzugeben. Speichern Sie diese Programme als Referenz und kopieren und modifizieren Sie sie für neue Programme, die Sie schreiben.

Anwendung: Untersuchung einer Funktion mit drei Variablen

Gehen wir davon aus, das wir das Programm für das Gesetz über Ideale Gase so modifizieren wollen, dass auch n für Mol als zusätzliche Variable eingeschlossen wird. Das bedeutet, das wir die folgende Funktion definieren

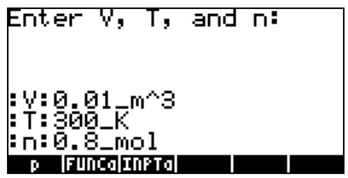
$$p(V,T,n) = (8.31451\_\frac{J}{K})\frac{n \cdot T}{V},$$

und modifizieren wollen, so dass drei verschiedene Variablen eingegeben werden können. Die Vorgehensweise ähnelt der, die wir schon bei der Definition der Funktion p(V.T) angewendet haben. Das Programm sieht dann so aus:

```
« "Enter V, T, and n:" {" ↵ :V:↵ :T:↵ :n:" {2 0} V }
INPUT OBJ→ →V T n '(8.31451_J/(K*mol))*(n*T/V)' »
```

Speichern Sie dieses Programm in der Variablen p.Um das Programm zu starten, drücken Sie p.

Geben Sie für V = 0.01_m^3, für T = 300_K und für n = 0,8_mol ein. Bevor Sie ENTER drücken, sieht der Stack folgendermaßen aus:



Drücken Sie ENTER. Das Resultat ist 199548.24_J/m^3, oder 199548.24_Pa = 199.55 kPa.

Eingabe über Eingabemasken

Die Funktion INFORM (← PRG NXT IN INFOR.) kann verwendet werden, um detaillierte Eingabemasken für ein Programm zu erstellen. Die Funktion INFORM benötigt 5 Argumente in nachstehender Reigenfolge:

1. Einen Titel Eine Zeichenkette, die die Eingabemaske beschreibt
2. Felddefinitionen: Eine Liste mit einer oder mehreren Felddefinitionen $\{s_1\ s_2\ \ldots\ s_n\}$, wobei jede Felddefinition, s_i, ein von zwei Formaten aufweisen kann:
   a. Eine einfache Feldbezeichnung: Eine Zeichenkette
   b. Eine Spezifikationsliste der Form {"label" "helpInfo" $type_0$ $type_1$ … $type_n$}. "label" ist eine Feldbezeichnung. "helpInfo" ist eine Zeichenkette, die die Feldbezeichnung im Detail beschreibt und die "Type"-Spezifikation ist eine Liste von Variablentypen,die in dem Feld erlaubt sind (siehe Objekttypen im Kapitel 24).

3. Information zum Feldformat: Eine einzelne Zahl *col* oder eine Liste {*col tabs*}. In dieser Spezifikation ist *col* die Anzahl der Spalten in der Eingabebox und *tabs* (optional) legt die Anzahl der Tab-Stops den Labeln und den Felder der Maske fest. Die Liste kann eine leere Liste sein. Vorgabewert sind *col* = 1 und *tabs* = 3.
4. Liste mit Reset-Werten: Eine Liste, die Werte enthält, welche die verschiedenen Felder beim Einsatz einer Eingabemaske zurücksetzen, wenn die Option RESET gewählt wird.
5. Liste mit Anfangs-Werten: Eine Liste, welche die Anfangs-Werte der Felder enthält.

Die Listen unter 4 und 5 können leer sein. Außerdem können Sie, wenn für diese Optionen keine Werte gewählt sind, den Befehl NOVAL (← PRG NXT IN NOVAL) verwenden.

Wenn die Funktion INFORM aktiviert wurde, erhalten Sie, wenn die Option CANCEL eingegeben wurde, eine Null oder eine Liste mit den in die Felder eingegebenen Werte in der spezifizierten Reihenfolge und die Zahl 1:

```
2:    {v1 v2 … vn}
1:               1
```

Wenn also der Wert in der Stack-Ebene gleich 0 ist, wurde keine Eingabe vorgenommen, während eine 1 anzeigt, dass die Eingabewerte in der Stack-Ebene 2 zur Verfügung stehen.

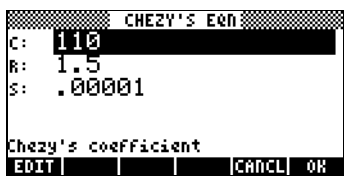
Beispiel 1 - Als Beispiel betrachten Sie das folgende Programm INFP1 (INput Form Program 1), um den Durchfluss Q in einem offenen Kanal nach der Chezy-Formel zu berechnen: $Q = C\cdot(R\cdot S)^{1/2}$, wobei C der Chezy-Koeffizient, eine Funktion aus der Rauhigkeit der Kanaloberfläche (typische Werte 80 - 150), ist, R ist der hydraulische Radius des Kanals (eine Länge) und S die Neigung des Kanalbetts (eine dimensionslose Zahl von 0.01 bis 0.000001). Das folgende Programm definiert eine Eingabemaske über die Funktion INFORM:

```
« " CHEZY'S EQN" { { "C:" "Chezy's coefficient" 0} { "R:"
"Hydraulic radius" 0 } { "S:" "Channel bed slope" 0} } { }
{ 120 1 .0001} { 110 1.5 .00001 }  INFORM »
```

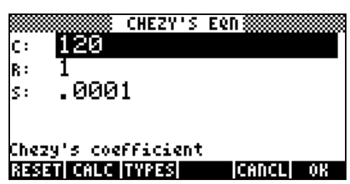
In dem Programm finden wird wie folgt die oben angeführten 5 Komponenten:

1. Titel: `" CHEZY'S EQN"`
2. Felddefinitionen: Hier haben wir drei mit den Labeln "C:", "R:", "S:", die Infostrings "Chezy coefficient", "Hydraulic radius", "Channel bed slope", und die Beschränkung der Eingabe auf den Datentyp 0 (Reale Zahlen) für alle drei Felder:
   `{ { "C:" "Chezy's coefficient" 0}`
   `{ "R:" "Hydraulic radius" 0 }`
   `{ "S:" "Channel bed slope" 0} }`
3. Information zum Feldformat: `{ }` (eine leere Liste, es werden die Vorgabewerte benutzt)
4. Liste mit Reset-Werten: `{ 120 1 .0001}`
5. Liste mit Anfangs-Werten: `{ 110 1.5 .00001}`

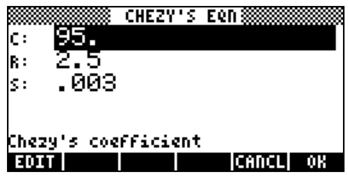
Speichern Sie dieses Programm in die Variable INFP1. Drücken Sie INFP1, um das Programm zu starten. Die Eingabemaske mit den geladenen Anfangswerten sieht wie folgt aus:



Um zu verdeutlichen, was mit diesen Werten beim Zurücksetzen geschieht, drücken Sie NXT RESET (wählen Sie *Reset all*, um die Werte der Felder zurück zu setzen):



Geben Sie nun verschiedene Werte für die drei Felder ein, sagen wir C = 95, R = 2,5 und S = 0,003. Drücken Sie nach jedem neuen Wert OK. Nach dieser Änderung sieht die Eingabemaske folgendermaßen aus:



Um nun die Werte in das Programm zu übertragen drücken Sie noch einmal OK. Hierdurch wird die Funktion INFORM aktiviert und im Stack erscheint:

Soweit die Demonstration der Verwendung von INFORM. Um zu sehen, wie diese Eingabewerte in einer Berechnung verwendet werden können, modifizieren wir das Programm wie folgt:

```
« " CHEZY'S EQN" { { "C:" "Chezy's coefficient" 0} { "R:"
"Hydraulic radius" 0 } { "S:" "Channel bed slope" 0} } { }
{ 120 1 .0001} { 110 1.5 .00001 }  INFORM IF THEN OBJ→ DROP →
C R S 'C*(R*S)' →NUM "Q" →TAG ELSE "Operation cancelled"
MSGBOX END »
```

Die oben gezeigten Programmschritte zeigen nach dem Befehl INFORM eine bedingte Verzweigung mit Hilfe von IF-THEN-ELSE-END (eine detaillierte Beschreibung finden Sie an anderer Stelle in diesem Kapitel). Das Programm kann abhängig vom Wert in der Stack-Ebene 1 zu einer von zwei Möglichkeiten springen. Ist der Wert 1, springt das Programm zu den Befehlen:

```
OBJ→ DROP → C R S 'C*√(R*S)' →NUM "Q" →TAG
```

Diese Befehle errechnen Q und setzen einen Tag auf Q. Befindet sich aber in der Stack-Ebene 1 der Wert 0 (was passiert wenn in der Eingabebox CANCEL eingegeben wird) , springt das Programm zu den Befehlen:

```
“Operation cancelled” MSGBOX
```

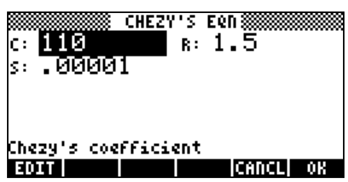
Diese Befehlen erzeugen eine Meldung, dass die Operation abgebrochen wurde.

Anmerkung: Die Funktion MSGBOX gehört zu einer Sammlung von Ausgabefunktionen im Menü PRG/OUT. Die Befehle Commands IF, THEN, ELSE, END finden Sie im Menü PRG/BRCH/IF. Die Funktionen OBJ→, →TAG befinden sich im Menü PRG/TYPE. DROP steht unter PRG/STACK zur Verfügung. Die Funktionen → und →NUM finden Sie auf der Tastatur.

Beispiel 2 – Um die Verwendung der 3. Funktion (Informationen zum Feldformat) der Argumente der Funktion INFORM zu illustrieren, ändern Sie die leere Liste des Programms INFP1 in { 2 1 }. Dies bedeutet, dass nur 2 Spalten anstelle der Standardvorgabe 3 und nur 1 Tab-Stop verwendet werden. Speichern Sie dieses neue Programm in die Variable INFP2:

```
« “ CHEZY’S EQN” { { “C:” “Chezy’s coefficient” 0} { “R:”
“Hydraulic radius” 0 } { “S:” “Channel bed slope” 0} } { 2 1 }
{ 120 1 .0001} { 110 1.5 .00001 }  INFORM IF THEN OBJ→ DROP →
C R S ‘C*(R*S)’ →NUM “Q” →TAG ELSE “Operation cancelled”
MSGBOX END »
```

Starten Sie das Programm INFP2, erhalten Sie die folgende Eingabemaske:



Beispiel 3 - Ändern Sie das Feldformat in { 3 0 } und speichern Sie das Programm als INFP3. Starten Sie das Programm und schauen Sie sich die neue Eingabemaske an:

Erstellen einer Auswahlbox

Die Funktion CHOOSE (←PRG NXT IN CHOOS) bietet dem Anwender die Möglichkeit, eine Auswahlbox in ein Programm zu integrieren. Diese Funktion benötigt drei Argumente:

1. Ein Prompt (eine Zeichenkette, die die Auswahlbox beschreibt
2. Eine Liste der Auswahlmöglichkeiten $\{c_1\ \ c_2\ \ \ldots\ \ c_n\}$. Eine Auswahlmöglichkeit c_i kann eine von zwei Formaten haben:
    a. Ein Objekt, z.B. eine Zahl, ein algebraischer Ausdruck, das in der Auswahlbox angezeigt wird und als Auswahlergebnis übernommen wird.
    b. Eine Liste {object_displayed object_result}, die so aufgebaut ist, dass object_displayed in der Auswahlbox angezeigt wird und in object_result wird dann das Auswahlergebnis übernommen.
3. Die Position eine Vorgabewertes in der Auswahlliste wird durch eine Nummer angezeigt. Ist diese 0, wird kein Vorgabewert hervorgehoben.

Die Aktivierung der CHOOSE Funktion gibt entweder Null zurück, wenn CANCEL gewählt wurde oder, wenn eine Auswahl getroffen wurde die Auswahl (z.B. v) und eine 1:

```
2:           v
1:           1
```

Beispiel 1 – Die Manning-Gleichung zur Kalkulation der Durchfluss-geschwindigkeit in einem offenen Kanal verwendet einen Koeffizienten C_u, der von den verwendeten Einheiten abhängt. Verwenden Sie S.I. (Systeme International), dann ist C_u = 1.0, Verwenden Sie jedoch E.S. (English System), dann ist C_u = 1.486. In dem folgenden Programm können Sie die Auswahlbox dazu benutzen C_u aufgrund der Auswahl des Maßsystems festzulegen. Speichern Sie es in die Variable CHP1 (Choose-Programm 1):

```
« "Units coefficient" { { "S.I. units" 1}
{ "E.S. units" 1.486}  } 1 CHOOSE »
```

Starten Sie das Programm (drücken Sie CHP2), so erscheint die folgende Auswahlbox:

Abhängig davon, ob Sie `S.I. units` oder `E.S. units gewählt haben legt` CHOOSE 1 oder 1.486 im Stack ab. Brechen Sie Choose-Box ab, gibt CHOISE eine Null (0) zurück.

Der Wert, der durch die Funktion CHOOSE zurückgegeben wird, kann von anderen Programmbefehlen, wie im modifizierten Programm CHP2 gezeigt, verwendet werden.

```
« "Units coefficient" { { "S.I. units" 1} { "E.S. units"
1.486}  } 1 CHOOSE IF THEN "Cu" →TAG ELSE "Operation
cancelled" MSGBOX END »
```

Mit Hilfe von IF-THEN-ELSE-END treffen in diesem Programm die Befehle nach CHOOSE eine Entscheidung anhand des in der Stack-Ebene 1 abgelegten Werts. Ist der Wert in der Stack-Ebene 1 gleich 1, erzeugt die Befehle `"Cu" →TAG` ein markiertes Ergebnis auf dem Bildschirm. Ist der Wert 0 erscheint bewirkt `"Operation cancelled" MSGBOX`, dass auf dem Bildschirm angezeigt wird, dass die Operation abgebrochen wurde.

Identifizieren von Programmausgaben.

Der einfachste Weg numerische Programmausgaben zu identifizieren ist, die Ergebnisse zu markieren. Bei einem Tag handelt es sich einfach um einen String, der an eine Zahl oder an ein Objekt angehängt wird. Der String bekommt einen Namen, der dem Objekt entspricht. Als wir beispielsweise weiter oben in den Programmen INTPa (oder INPT1) und INPT2 nach Fehlern gesucht haben, erhielten wir als Ergebnisse numerische Ausgaben wie `:a:35`.

Markieren eines numerischen Ergebnisses

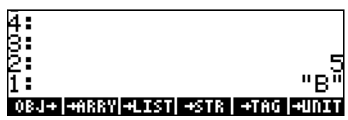
Um ein numerisches Ergebnis zu markieren, müssen Sie Zahl und Markierungs-String in die Stack-Ebene 2 ablegen, dann verwenden Sie die

Funktion →TAG (← PRG TYPE →TAG) . Um beispielsweise das Ergebnis `B:5.`zu erhalten, geben Sie folgendes ein:

5 ENTER → "" ALPHA B ← PRG TYPE →TAG

Auflösen eines markierten Ergebnisses in eine Zahl und einen Tag

Um ein markiertes Ergebnis in die Zahl und den zugehörigen Tag aufzulösen verwenden Sie die Funktion OBJ→ (← PRG TYPE OBJ→). Durch →OBJ bleibt der numerische Wert in der Stack-Ebene 2 und der Tag wird in die Stack-Ebene 1 abgelegt. Wollen Sie nur den numerischen Wert weiter verwenden, können Sie den Tag mit Hilfe der Rücktaste ◄ löschen. So produziert beispielsweise die Auflösung der Menge B:5 (siehe oben) folgendes Ergebnis:



"Extrahieren" einer markierten Menge

"Extrahieren" meint, herausziehen des Objekts aus einer Markierten Menge. Die Funktion wird mit der nachstehenden Tastenfolge aufgerufen: ← PRG TYPE NXT DTAG. So gibt DTAG beispielsweise bei einer markierten Menge `a:2` den Wert 2 zurück.

Anmerkung: Bei mathematischen Operationen mit markierten Mengen extrahiert der Rechner die numerischen Werte automatisch. So zeigt beispielsweise die linke Abbildung zwei markierte Mengen vor und die rechte nachdem im RPN-Modus × gedrückt wurde:

Beispiele für markierte Ausgaben

Beispiel 1 Markierte Ausgabe mit FUNCa

Wir wollen die vorher definierte Funktion FUNCa so ändern, dass sie markierte Ausgaben produziert.Rufen Sie mit ↱ FUNCa den Inhalt von FUNCa in den Stack. Die Originalfunktion sieht folgendermaßen aus:

```
« "Enter a:" {"↵:a:" {2 0} V } INPUT OBJ→ → a «
              '2*a^2+3' →NUM »»
```

Modifizieren Sie sie wie folgt:

```
« "Enter a:" {"↵:a:" {2 0} V } INPUT OBJ→ → a «
          '2*a^2+3' →NUM "F" →TAG »»
```

Speichern Sie das modifizierte Programm wieder mit Hilfe von ↰ FUNCa unter FUNCa. Starten Sie das Programm, indem Sie FUNCa drücken. Geben Sie, wenn Sie dazu aufgefordert werden den Wert 2 ein, und drücken ENTER. Das Resultat ist nun ein markiertes Ergebnis `F:11` .

Beispiel 2 - Markierte Ein- und Ausgabe mit FUNCa

In diesem Beispiel modifizieren wir FUNCa so, dass nicht nur das berechnete Ergebnis sondern auch eine Kopie der Eingabe markiert ausgegeben wird.

Rufen Sie mit ↱ FUNCa den Inhalt von FUNCa in den Stack.

```
« "Enter a:" {"↵:a:" {2 0} V } INPUT OBJ→ → a «
          '2*a^2+3' →NUM "F" →TAG » »
```

Modifizieren Sie sie wie folgt:

```
« "Enter a:" {"↵:a:" {2 0} V } INPUT OBJ→ → a «
       '2*a^2+3' EVAL "F" →TAG a SWAP» »
```

(Beachten Sie, dass die Funktion SWAP über die Tastenfolge ↰ PRG STACK SWAP) aufgerufen wird. Speichern Sie das modifizierte Programm wieder mit Hilfe von ↰ FUNCa unter FUNCa. Starten Sie das Programm, indem Sie FUNCa drücken. Geben Sie, wenn Sie dazu aufgefordert werden den Wert 2 ein, und drücken ENTER. Als Ergebnis erhalten Sie die markierten Zahlen `a:2.` in der Stack-Ebene 2 und `F:11.` in der Stack-Ebene 1.

Anmerkung: Da wir einen Eingabestring für die Eingabe der Daten verwenden, speichert die lokale Variable a einen markierten Wert) in obigem Beispiel `:a:2`). Daher müssen wir die Eingabe nicht markieren. Wir müssen nur in das Unterprogramm ein `a` vor die SWAP-Funktion einfügen und die Markierte Eingabe wird in den Stack geschrieben. Um es noch einmal klarzustellen, bei der Kalkulation einer Funktion wird die Markierung der Eingabe automatisch fallen gelassen und nur der numerische Wert zur Kalkulation benutzt.

Um sich die Funktion von FUNCa Schritt für Schritt anzuschauen, können Sie die Funktion DBUG wie folgt verwenden:

' FUNCa ENTER	Kopiert den Programmnamen in die Stack-Ebene 1
← PRG NXT NXT RUN DBG	Startet die Fehlersuche (den Debugger)
SST↓	Fehlersuche Schritt für Schritt, Ergebnis: "Enter a:"
SST↓	Ergebnis: {" ↵a:" {2 0} V}
SST↓	Ergebnis: Anwender wird aufgefordert, für a einen Wert einzugeben
2 ENTER	Geben Sie für a 2 ein. Ergebnis: "↵:a:2"
SST↓	Ergebnis: a:2
SST↓	Ergebnis: Stack leeren, ausführen von →a
SST↓	Ergebnis: Stack leeren, ins Unterprogramm springen «
SST↓	Ergebnis: '2*a^2+3'
SST↓	Ergebnis: Stack leeren, ins Unterprogramm springen
SST↓	Ergebnis: 11.,
SST↓	Ergebnis: "F"
SST↓	Ergebnis: F: 11.
SST↓	Ergebnis: a:2.
SST↓	Ergebnis: Austausch von Ebene 1 und 2
SST↓	Unterprogramm verlassen »
SST↓	Hauptprogramm verlassen »

Beispiel 3 - Markierte Ein- und Ausgabe mit Funktion p(V,T)

In diesem Beispiel modifizieren wir das Programm ▒▒ so, Eingabe, Ausgabe und Ergebnis markiert werden. Rufen Sie mit ↱▒▒ den Inhalt des Programms in den Stack.

```
« "Enter V, T, and n:" {"↵ :V:↵ :T:↵ :n:" {2 0} V }
INPUT OBJ→ →V T n '(8.34451_J/(K*mol))*(n*T/V)' »
```

Modifizieren Sie sie wie folgt:

```
« "Enter V, T and n: " {"↵ :V:↵ :T: " {2 0} V } INPUT OBJ→
→V T n « V T n'(8.34451_J/(K*mol))*(n*T/V)' EVAL "p" →TAG
» »
```

Anmerkung: Beachten Sie, dass wir die Berechnung und Markierung der Funktion p(V,T,n) in ein Unterprogramm gelegt [die Befehlssequenz die sich innerhalb der inneren Programmsymbole « » befindet] und den Aufruf der Variablen V, T und n vorangestellt haben. Dies ist erforderlich, da ohne die Trennung der beiden Variablenlisten (`V T N « V T n`) durch Programmsymbole das Programm davon ausgeht, das der Eingabebefehl

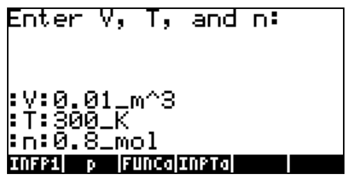
`→V T N V T n`

sechs Eingabewerte erfordert, obwohl nur drei zur Verfügung stehen.Es würde zu einer Fehlermeldung kommen und die Programmausführung würde abgebrochen.

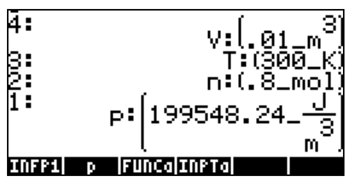
Um das Unterprogramm in die modifizierte Version des Programms ▒▒ einzufügen, müssen wir sowohl am Anfang als auch am Ende des Unterprogramms ↱ « » verwenden. Da das Programmsymbol im als Paar erscheint ↱ « » wenn es aufgerufen wird, müssen wir das Abschlusssymbol am Anfang (») und das Anfangssymbol am Ende («) des Unterprogramms löschen.

Um ein irgendein Zeichen während der Bearbeitung eines Programms zu löschen, setzen Sie den Cursor rechts hinter das Zeichen und löschen Sie es mit der Taste ⬅.

Speichern Sie das Programm mit der Tastenfolge ↰ p in die Variable p. Starten Sie das Programm, indem Sie p drücken. Geben Sie, wenn Sie dazu aufgefordert werden, für V = 0.01_m^3, für T = 300_K und für n = 0,8_mol ein. Bevor Sie ENTER drücken, sieht der Stack folgendermaßen aus:



Nach Ausführung des Programms erscheint der Stack wie folgt:



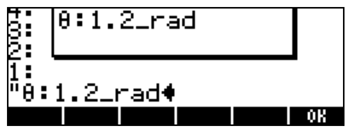
Zusammenfassung: Die Verwendung von Markierungen, um Ein- und Ausgabevariable zu identifizieren, zieht sich wie ein Faden durch alle drei Beispiele. Verwenden wir einen Eingabe-String für den Eingabewert, werden diese vormarkiert und können einfach wieder zur Ausgabe in den Stack geladen werden. Der Befehl →TAG erlaubt es uns, die Ausgabe eines Programms zu identifizieren.

Verwendung von Meldungen

Eine professionellere Art, die Ausgabe eines Programms anzuzeigen, ist der Einsatz von Meldefenstern. Der Befehl für ein Meldefenster wird durch die Tastenfolge ↰ PRG NXT OUT MSGBOX aufgerufen. Um den Ausgabe-String in einem Fenster anzuzeigen, muss er sich in der Stack-Ebene 1 befinden. Um die Funktion des Befehls MSGBOX kennen zu lernen, versuchen Sie folgendes:

[→] "" [ALPHA] [→] [T] [ALPHA] [←] :: [1] [.] [2]
[→] _ [ALPHA] [←] [R] [ALPHA] [←] [A] [ALPHA] [←] [D]
[←] PRG [NXT] OUT MSGBOX

Als Ergebnis sehen Sie folgendes Meldefenster:



Drücken Sie OK, um das Meldefenster zu löschen.

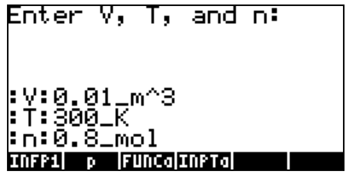
Sie können ein Meldefenster zur Anzeige einer markierten Ausgabe verwenden, indem Sie diese in einen Ausgabe-String umwandeln. Um ein Markiertes Ergebnis, eine Formel oder einen nicht markierten Wert in einen String umzuwandeln, verwenden Sie die Funktion →STR, verfügbar über die Tastenfolge [←] PRG TYPE →STR.

Verwenden eines Meldefenster zur Programmausgabe

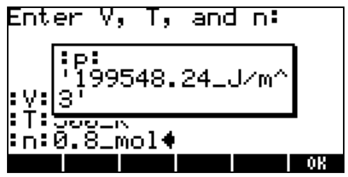
Die Funktion p aus dem letzten Beispiel kann folgendermaßen geändert werden:

```
« "Enter V, T and n: " {"↵ :V:↵ :T:↵ :n: " {2 0} V }
INPUT OBJ→ →V T n « V T n
'(8.34451_J/(K*mol))*(n*T/V)' EVAL "p" →TAG →STR MSGBOX »
»
```

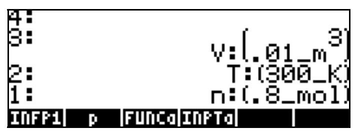
Speichern Sie das Programm mit der Tastenfolge [←] p in die Variable p. Starten Sie das Programm, indem Sie p drücken. Geben Sie, wenn Sie dazu aufgefordert werden, für V = 0.01_m^3, für T = 300_K und für n = 0,8_mol ein.
Wie in der älteren Version von p sieht der Stack, bevor Sie [ENTER] drücken, folgendermaßen aus:



Bei der ersten Programmausgabe handelt es sich um ein Meldefenster mit folgendem String:



Drücken Sie OK, um das Meldefenster zu löschen.Im Stack steht nun folgendes:



Ein- und Ausgabe in einem Meldefenster anzeigen

Wir können das Programm so modifizieren, dass sowohl Ein- als auch Ausgabe im Meldefenster angezeigt werden. Das dahingehend abgeänderte Programm p sieht folgendermaßen aus:

```
« "Enter V, T and n: " {"↵ :V:↵ :T:↵ :n: " {2 0} V } INPUT
OBJ→ → V T n « V →STR "↵ " + T →STR "↵ " + n →STR "↵ " +
'(8.34451_J/(K*mol))*(n*T/V)' EVAL "p" →TAG →STR + + +
MSGBOX » »
```

Beachten Sie, dass Sie im Unterprogramm nach den Variablennamen V, T und n folgenden Code eingeben:

```
→STR "↵ " +
```

Um diesen Code das erste Mal einzugeben, verwenden Sie die Tastenfolge:

PRG TYPE →STR "" ↵ ▶ +

Da die Funktionen des Menüs TYPE über die Funktionstasten weiter verfügbar sind, geben Sie im Unterprogramm für die zweite und dritte Code-Folge (→STR "↵ " +) (also nach den Variablen T und n) die nachstehende Tastenfolge ein:

→STR "" ↵ ▶ +

Sie werden sehen, dass nach der Tastenfolge ↵ im Stack eine neue Zeile erzeugt wird.

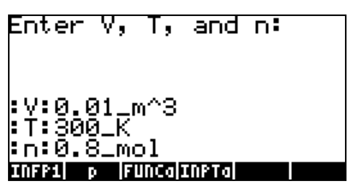
Die letzte Modifikation, die noch vorgenommen werden muss, ist die Eingabe von drei Pluszeichen nach dem Funktionsaufruf am Ende des Unterprogramms.

> **Anmerkung**: Die Pluszeichen (+) dienen in diesem Programm zur *Verknüpfung* der Strings. Verknüpfung bedeutet einfach die Zusammenführung einzelner Strings.

Um das Programm auszuführen,

- Speichern Sie das Programm mit der Tastenfolge p in die Variable p.
- Starten Sie das Programm, indem Sie p drücken.
- Geben Sie, wenn Sie dazu aufgefordert werden, für V = 0.01_m^3, für T = 300_K und für n = 0,8_mol ein.

Wie in der älteren Version von [p] sieht der Stack, bevor Sie [ENTER] drücken, folgendermaßen aus:



Bei der ersten Programmausgabe handelt es sich um ein Meldefenster mit folgendem String:

Drücken Sie OK, um das Meldefenster zu löschen.

Integrieren von Einheiten in ein Programm

Wie Sie bei den verschiedenen Versionen des Programms p gesehen haben, ist das Anhängen von Einheiten an die Eingabewerte eine langwierige Sache.

Sie können es so einrichten, dass das Programm selbst diese Einheiten an die Ein- und Ausgabewerte anhängt. Wir demonstrieren dies, indem wir unser Programm p einweiteres Mal wie folgt ändern.

Laden Sie den Inhalt des Programms p wieder mit ↱ p in den Stack und modifizieren es wie folgt:

> **Anmerkung**: Um es einfacher lesen zu können, haben wir das Programm willkürlich in verschiedene Zeilen aufgeteilt. Das Programm muss nicht unbedingt so im Stack angezeigt werden. Die Befehlsfolge ist auf jeden Fall richtig. Beachten Sie auch, dass das Zeichen ↵ nicht im Stack angezeigt wird, sondern eine neue Zeile erzeugt.

```
« “Enter V,T,n [S.I.]: “ {“↵ :V:↵ :T:↵ :n: “ {2 0} V }
INPUT OBJ→ →V T n
« V ‘1_m^3’ *  T ‘1_K’ *  n ‘1_mol’ *  →V T n
« V “V” →TAG →STR “↵ ” + T “T” →TAG →STR “↵ ” +  n
“n” →TAG →STR “↵ ” +
‘(8.34451_J/(K*mol))*(n*T/V)‘ EVAL “p” →TAG →STR + + +
MSGBOX » » »
```

Diese neue Programmversion enthält eine zusätzliche Unterprogrammebene (also eine dritte Ebene der Programmsymbole « » und einige Schritte, die Listen verwenden, beispielsweise

`V '1_m^3' * { } + T '1_K' * + n '1_mol' * + EVAL → V T n`

Dieser Programmcode wird wie folgt *interpretiert*: (Wir verwenden als Eingabewerte `:V:0.01`, `:T:300`, und `:n:0.8`):

1. `V` : Der Wert von V als markierte Eingabe (z.B. V:0.01) wird in den Stack gestellt.

2. `'1_m^3'` : Die S.I.-Einheiten von V werden dann in die Stack-Ebene 1 gestellt und die markierte Eingabe für V in die Stack-Ebene 2 verschoben.

3. `*` : Indem wir die Inhalte der Stack-Ebenen 1 und 2 multiplizieren, erzeugen wir eine Zahl mit Einheiten (z.B., 0.01_m^3), aber der Tag geht verloren.

4. `T '1_K' *` : Kalkulation von T mit S.I.-Einheiten

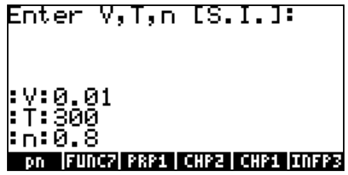
5. `n '1_mol' *` : Kalkulation von n mit S.I.-Einheiten

6. `→ V T n` : Die Werte von V, T und n in den Stack-Ebenen 3, 2 und 1 werden an die nächste Unterprogrammebene weitergegeben.

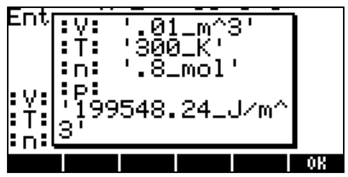
Um zu sehen, wie diese Programmversion arbeitet, machen Sie folgendes:

- Speichern Sie das Programm mit der Tastenfolge ⬅ in die Variable p.
- Starten Sie das Programm, indem Sie drücken.
- Geben Sie, wenn Sie dazu aufgefordert werden, für V = 0.01, für T = 300 und für n = 0.8 ein (Einheiten sind nicht mehr erforderlich).

Bevor Sie ENTER drücken, sieht der Stack folgendermaßen aus:



Drücken Sie ENTER, um das Programm auszuführen. Bei der ersten Programmausgabe handelt es sich um ein Meldefenster mit folgendem String:



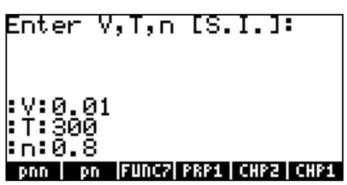
Drücken Sie OK, um das Meldefenster zu löschen.

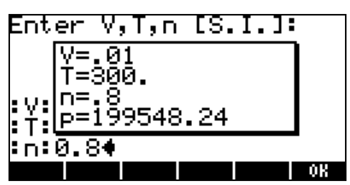
Ausgabe im Meldefenster ohne Einheiten

Wir wollen das Programm ein weiteres Mal ändern, um es ganz ohne Einheiten zu verwenden. Das Programm sieht dann so aus:

```
« "Enter V,T,n [S.I.]:" {"↵ :V:↵ :T:↵ :n: " {2 0} V }
INPUT OBJ→ → V T n
« V DTAG  T DTAG  n DTAG →  V T n
« "V=" V →STR + "↵ "+ "T=" T →STR + "↵ "+ "n="  n →STR +
"↵ "+
'8.34451*n*T/V' EVAL →STR "p=" SWAP + + + + MSGBOX » » »
```

Wird das Programm mit den Werten V = 0.01, T = 300 und n = 0.08 gestartet, erscheint im Meldefenster folgende Ausgabe:





Drücken Sie OK, um das Meldefenster zu löschen.

Relationale und logische Operatoren

Bis jetzt haben wir hauptsächlich mit sequentiellen Programmen gearbeitet. Die RPL-Sprache bietet Befehle, die im Programmablauf Verzweigungen und Schleifen ermöglichen. Viele davon treffen Entscheidungen basierend darauf, ob ein logische Aussage wahr oder unwahr ist. In diesem Absatz stellen wir relationale und logische Operatoren vor, mit deren Hilfe solche logischen Aussagen aufgebaut werden können.

Relationale Operatoren

Relationale Operatoren vergleichen die relative Position von zwei Objekten. Wenn Sie beispielsweise mit Realzahlen arbeiten, werden mit relationalen Operatoren die relativen Positionen von zwei oder mehreren Realzahlen ermittelt. Abhängig von den aktuellen Zahlen können solche Aussagen wahr (im Rechner repräsentiert durch die Zahl 1) oder falsch (im Rechner repräsentiert durch die Zahl 0) sein.

Für die Programmierung stehen folgende relationalen Operatoren zur Verfügung:

Operator	**Bedeutung**	**Beispiel**
==	„ist gleich“	‘x==2’
≠	„ist nicht gleich“	‘3 ≠ 2’
<	„ist kleiner als“	‘m<n’
>	„ist größer als“	‘10>a’
≥	„ist größer als oder gleich“	‘p ≥ q’
≤	„ist kleiner als oder gleich“	‘7≤12’

Alle Operatoren, mit Ausnahme von == (der durch die Tastenfolge ↱ = ↱ = erzeugt wird), stehen auf der Tastatur zur Verfügung. Sie finden sie auch in ↰ PRG TEST.

Zwei Zahlen, Variablen oder algebraische Ausdrücke verknüpft mit einem relationalen Operator bilden einen logischen Ausdruck, der wahr (1), falsch (0) oder nicht abgefragt sein kann. Um festzustellen, ob ein logischer Ausdruck wahr oder unwahr ist, stellen Sie ihn in die Stack-Ebene 1 und drücken (EVAL). Beispiele:

‘2<10’ EVAL, Ergebnis: 1. (wahr)

‘2<10’ EVAL, Ergebnis: 0. (unwahr)

In dem nächsten Beispiel gehen wir davon aus, dass die Variable m nicht initialisiert wurde (ihr wurde noch kein numerischer Wert zugeordnet);

‘2==m’ EVAL, Ergebnis: ‘2==m’

Die Tatsache, dass das Ergebnis nach dem Vergleich der Originalaussage entspricht, bedeutet, dass keine eindeutige Aussage getroffen werden kann.

Logische Operatoren

Logische Operatoren sind Ausdrücke, die in einfache logische Aussagen integriert werden oder diese modifizieren können. Die verfügbaren logischen Operatoren können über die nachstehende Tastenfolge aufgerufen werden: ← PRG TEST NXT.

Folgende Operatoren stehen zur Verfügung: AND, OR, XOR (exklusiv oder), NOT und SAME. Die Operatoren erzeugen Ergebnisse die wahr oder unwahr sind, abhängig vom Wert der betroffenen logischen Aussage. Der Operator NOT (Negation) gilt für einzelne logische Aussagen. Alle anderen gelten immer für zwei logische Aussagen.

Eine Auflistung aller möglichen Kombinationen für eine oder zwei Aussagen zusammen mit dem Ergebnis durch Anwendung eines bestimmten logischen Operators führt zu einer Wahrheitstabelle für den Operator. Nachfolgend finden Sie Wahrheitstabellen für alle im Rechner verfügbaren logischen Operatoren:

p	NOT p
1	0
0	1

p	q	p AND q
1	1	1
1	0	0
0	1	0
0	0	0

p	q	p OR q
1	1	1
1	0	1
0	1	1
0	0	0

p	q	p XOR q
1	1	0
1	0	1
0	1	1
0	0	0

Zusätzlich ist im Rechner der logische Operator SAME integriert Hierbei handelt es sich nicht um einen Standardoperator. Er stellt fest, ob zwei Objekte identisch sind. Ist dies der Fall, wird 1 (wahr) zurückgegeben, ist dies nicht der Fall, dann wird 0 (unwahr) zurückgegeben. Geben Sie beispielsweise folgende Übung im RPN-Modus ein, wird als Wert 0 zurückgegeben:

‘SQ(2)’ ENTER 4 ENTER SAME

Beachten Sie, dass bei der Verwendung von SAME der Begriff "Identisch" sehr eng ausgelegt wird. Aus diesem Grund ist SQ(2) nicht identisch mit 4, obwohl sich bei beiden ein Zahlenwert von 4 ergibt.

Programmverzweigung

Verzweigung eines Programmablaufes bedeutet, dass sich das Programm zwischen zwei oder mehr Abläufen entscheidet. Die RPL-Sprache bietet verschiedene Befehle, die zu diesem Zweck verwendet werden können. Das Menü mit diesen Befehlen wird über die nachstehende Tastenfolge aufgerufen:

Das Menü zeigt Untermenüs für die Befehlskonstruktionen

Die Konstruktionen IF…THEN..ELSE…END und CASE…THEN…END werden als Verzweigungsbefehle bezeichnet. Die weiteren Konstruktionen wie START, FOR, DO und WHILE dienen dazu, Wiederholungen in ein Programm einzufügen und werden als Schleifenbefehle bezeichnet. Die letzten Konstruktionen werden in einem späteren Absatz im Detail behandelt.

Verzweigung mit IF

In diesem Absatz zeigen wir Beispiele mit den Befehlen IF…THEN…END und IF…THEN…ELSE…END.

Der Befehl IF…THEN…END

IF…THEN…END ist die einfachste Form des IF-Befehls. Die allgemeine Form des Befehls sieht wie folgt aus:

`IF` *`logical_statement`* `THEN` *`program_statements`* `END`.

Dieser Befehl arbeitet wie folgt:

1. Ermitteln der logischen Aussage(logical_statement)
2. Ist die logische Aussage wahr, ausführen der Programmschritte (program_statements) und Fortsetzung des Programmablaufes nach dem Befehl END.

3. Ist die logische Aussage unwahr, überspringen der Programmschritte und Fortsetzung des Programmablaufes nach dem Befehl END.

Um IF, THEN, ELSE und END einzugeben, verwenden Sie:

← PRG BRCH IF

Die Funktionen IF THEN ELSE END stehen in dem Menü zur Verfügung und müssen vom Anwender einzeln eingegeben werden. Alternativ können Sie die Konstruktion IF…THEN…END mit der nachstehenden Tastenfolge direkt in den Stack eingeben:

← PRG BRCH ← IF

Hierdurch wird im Stack folgende Eingabe abgelegt:

Der Cursor ⇐ befindet sich vor dem Befehl IF und der Anwender muss die logische Aussage eingeben, die den IF-Befehl während des Programmablaufs aktiviert.
Beispiel: Geben Sie das folgende Programm ein

```
« → x « IF 'x<3' THEN 'x^2' EVAL END "Done" MSGBOX » »
```

und speichern es unter dem Namen 'f1'. Drücken Sie VAR, um sich zu vergewissern, dass die Variable f1 tatsächlich in Ihrem Variablen-Menü zur Verfügung steht. Prüfen Sie die folgenden Ergebnisse:

0 f1 Ergebnis: 0
1.2 f1 Ergebnis: 1.44
3.5 f1 Ergebnis: keine Wirkung
10 f1 Ergebnis: keine Wirkung

Diese Ergebnisse bestätigen, dass die IF…THEN…END-Konstruktion korrekt arbeitet. Das Programm berechnet die Funktion $f_1(x) = x^2$, wenn x < 3 (anderenfalls erfolgt keine Ausgabe).

Der Befehl IF…THEN…ELSE…END

Die Befehlskonstruktion IF…THEN…ELSE…END bietet zwei verschiedene Programmabläufe, die von dem Wahrheitswert der logischen Aussage abhängen. Die allgemeine Form des Befehls sieht wie folgt aus:

```
IF logical_statement THEN program_statements_if_true
          ELSE program_statements_if_false END.
```

Dieser Befehl arbeitet wie folgt:

1. Ermitteln der logischen Aussage
2. Ist die logische Aussage wahr, ausführen der Programmschritte (program_statements_if_true) und Fortsetzung des Programmablaufes nach dem Befehl END.
3. Ist die logische Aussage unwahr, ausführen der Programmschritte (program_statements_if_false) und Fortsetzung des Programmablaufes nach dem Befehl END.

Alternativ können Sie die Konstruktion IF…THEN…ELSE…END mit der nachstehenden Tastenfolge direkt in den Stack eingeben:

← PRG BRCH → IF

Hierdurch wird im Stack folgende Eingabe abgelegt:

Beispiel: Geben Sie das folgende Programm ein:

```
« → x « IF 'x<3' THEN 'x^2' ELSE '1-x' END EVAL "Done" MSGBOX
» »
```

und speichern es unter dem Namen ‘f2’. Drücken Sie VAR, um sich zu vergewissern, dass die Variable f2 tatsächlich in Ihrem Variablen-Menü zur Verfügung steht. Prüfen Sie die folgenden Ergebnisse:

0 f2 Ergebnis: 0 1.2 f2 Ergebnis: 1.44
3.5 f2 Ergebnis: -2.5 10 f2 Ergebnis: -9

Diese Ergebnisse bestätigen, dass die IF…THEN…ELSE…END-Konstruktion korrekt arbeitet. Das Programm berechnet folgende Funktion:

$$f_2(x) = \begin{cases} x^2, if\ x < 3 \\ 1 - x, otherwise \end{cases}$$

Anmerkung: Für diesen speziellen Fall könnte als Alternative auch die Funktion IFTE in folgender Form verwendet werden: ‘f2(x) = IFTE(x<3,x^2,1-x)’

Verschachtelte IF…THEN…ELSE…END Konstruktionen

In den Meisten Programmiersprachen, die über eine IF…THEN…ELSE…END Konstruktion verfügen, sieht das allgemeine Format folgendermaßen aus:

```
IF logical_statement THEN
      program_statements_if_true
ELSE
      program_statements_if_false
END
```

Wenn Sie ein Programm mit einer IF-Konstruktion für den Rechner erstellen, können Sie zunächst den Code, wie oben dargestellt, von Hand notieren. Für Programm f2 könnten Sie beispielsweise schreiben

```
IF x<3 THEN
      x^2
ELSE
      1-x
END
```

Diese Konstruktion arbeitet gut, wenn Ihre Funktion nur zwei Verzweigungen hat. Bei Funktionen mit drei oder mehr Verzweigungen müssen Sie IF…THEN…ELSE…END Konstruktionen ineinander verschachteln. Betrachten Sie beispielsweise folgende Funktion:

$$f_3(x) = \begin{cases} x^2, \textit{if } x < 3 \\ 1 - x, \ \textit{if } 3 \le x < 5 \\ \sin(x), \textit{if } 5 \le x < 3\pi \\ \exp(x), \textit{if } 3\pi \le x < 15 \\ -2, \textit{elsewhere} \end{cases}$$

Dies wäre ein möglicher Weg, diese Funktion mit IF… THEN … ELSE … END Konstruktionen zu errechnen:

```
IF x<3 THEN
        x^2
ELSE
        IF x<5 THEN
                1-x
        ELSE
                IF x<3π         THEN
                        sin(x)
                ELSE
                        IF x<15 THEN
                                exp(x)
                        ELSE
                                -2
                        END
                END
        END
END
```

Eine komplexe IF-Konstruktion wie diese wird <u>verschachtelte</u> IF … THEN … ELSE … END Konstruktion genannt.

Eine Möglichkeit, F3(x) mit einer verschachtelten IF-Konstruktion zu untersuchen, wäre das folgende Programm:

```
« → x « IF 'x<3' THEN 'x^2' ELSE IF 'x<5' THEN '1-x' ELSE IF
'x<3*π' THEN 'SIN(x)' ELSE IF 'x<15' THEN 'EXP(x)' ELSE –2
END END END END EVAL » »
```

Speichern Sie das Programm in die Variable F3 und versuchen Sie folgendes:

1.5 F3	Ergebnis: 2.25 (d.h., x^2)
2.5 F3	Ergebnis: 6.25 (d.h., x^2)
4.2 F3	Ergebnis: -3.2 (d.h., 1-x)
5.6 F3	Ergebnis: -0.631266… (d.h., sin(x), mit x in Radian)
12 F3	Ergebnis: 162754.791419 (d.h., exp(x))
23 F3	Ergebnis: -2. (d.h., -2)

Der Befehl CASE

Der Befehl CASE wird verwendet, um verschiedene Programmverzweigungen zu programmieren, wie mit den verschachtelten IF-Konstruktionen oben. Die allgemeine Form des Befehls sieht wie folgt aus:

```
CASE
Logical_statement1 THEN program_statements1 END
Logical_statement2 THEN program_statements2 END
.
.
.
Logical_statement THEN program_statements END
Default_program_statements (optional)
END
```

Bei der Auswertung dieser Konstruktion testet das Programm jedes einzelne *logical_statements,* bis es eine findet, die wahr ist. Das Programm führt dann die entsprechenden *program_statements* aus und fährt mit dem Programm hinter dem Befehl END fort.

Die Befehle CASE, THEN und END finden Sie mit der Tastenfolge
← PRG BRCH CASE.

Wenn Sie sich im Menü BRCH befinden, d.h., (← PRG BRCH), können Sie die folgenden Tastenkürzel verwenden, um Ihre CASE-Konstruktion einzugeben (Die Position des Cursors wird durch das Symbol ⇐ angezeigt):

- ← CASE: Startet die Case-Konstruktion mit den Eingabeaufforderungen: CASE ⇐ THEN END END

- → CASE: Beendet eine CASE-Zeile, indem es THEN ⇐ END anfügt.

Beispiel – Program $f_3(x)$ mit CASE-Befehlen
Die Funktion wird durch die folgenden 5 Ausdrücke definiert:

$$f_3(x) = \begin{cases} x^2, \ if \ x < 3 \\ 1-x, \ if \ 3 \leq x < 5 \\ \sin(x), \ if \ 5 \leq x < 3\pi \\ \exp(x), if \ 3\pi \leq x < 15 \\ -2, \ elsewhere \end{cases}$$

Im RPN-Modus können wir die Funktion wie folgt eingeben:

```
« → x « CASE 'x<3' THEN 'x^2' END 'x<5' THEN '1-x' END
'x<3*π' THEN 'SIN(x)' END 'x<15' THEN 'EXP(x)' END –2 END
EVAL » »
```

Speichern Sie das Programm unter F3c. Versuchen Sie dann das folgende Beispiel:

1.5	F3c	Ergebnis: 2.25 (d.h., x^2)
2.5	F3c	Ergebnis: 6.25 (d.h., x^2)
4.2	F3c	Ergebnis: -3.2 (d.h., 1-x)

5.6 F3c Ergebnis: -0.631266… (d.h., sin(x), mit x in Radian)
12 F3c Ergebnis: 162754.791419 (d.h., exp(x))
23 F3c Ergebnis: -2. (d.h., -2)

Wie Sie sehen erzeugt F3c genau die gleichen Ergebnisse wie f3. Der einzige Unterschied liegt in den Verzweigungskonstruktionen. Bei der Funktion $f_3(x)$, die 5 verschiedene Ausdrücke zur Definition verwendet, ist die CASE-Konstruktion vielleicht einfacher zu codieren al seine Anzahl verschachtelter IF … THEN … ELSE … END Konstruktionen.

Programmschleifen

Bei Programmschleifen handelt es sich um Konstruktionen, die es dem Programm ermöglichen, eine Anzahl von Befehlen zu wiederholen. Nehmen Sie beispielsweise an, das Sie die Summation des Quadrates der Ganzzahlen von 0 bis n berechnen wollen, also

$$S = \sum_{k=0}^{n} k^2$$

Um diese Summation zu berechnen, müssen Sie im Gleichungs-Editor nur ↱ _Σ eingeben und dann die Grenzwerte und Ausdrücke für die Summation laden (Beispiele hierzu finden Sie in den Kapiteln 2 und 13). Um den Einsatz von Programmschleifen zu demonstrieren, werden wir diese Summation mit unserem eigenen RPL-Code berechnen. RPL bietet vier verschiedene Befehle zur Programmierung von Programmschleifen, START, FOR, DO und WHILE. Die Befehle START und FOR verwenden einen Index oder Zähler, um festzulegen, wie oft die Schleife ausgeführt werden soll. Die Befehle DO und WHILE verwenden eine logische Aussage, um zu entscheiden, wann eine Schleife verlassen wird. Die Verwendung der Schleifenbefehle wird im nächsten Absatz im Detail erklärt.

Der Befehl START

START verwendet zwei Werte eines Index, um eine Anzahl an Befehlen zu wiederholen. Es existieren zwei verschiedene START-Konstruktionen: START…NEXT und START … STEP. START…NEXT wird verwendet, wenn der

Index um 1 inkrementiert wird. START…STEP wird eingesetzt, wenn das Inkrement des Index vom Anwender festgelegt wird.

Die Befehle der START-Konstruktion rufen Sie wie folgt auf:

⬑ PRG BRCH START START

Innerhalb des Menüs BRCH (⬑ PRG BRCH) stehen folgende Tastenfolgen zur Verfügung, um eine START-Konstruktion zu erstellen (das Symbol zeigt die Cursor-Position):

- ⬑ START : Startet die START…NEXT Konstruktion: START ⬅ NEXT
- ⬏ START : Startet die START…STEP Konstruktion: START ⬅ STEP

Die START…NEXT Konstruktion

Die allgemeine Form des Befehls sieht wie folgt aus:

```
start_value end_value START program_statements NEXT
```

Da in diesem Fall das Inkrement 1 ist, sollten Sie sicherstellen, dass `start_value < end_value`, damit die Schleife beendet werden kann. Anderenfalls kommt es zu einer Endlosschleife.

Beispiel – Berechnung der oben definierten Summation S

Die START…NEXT Konstruktion hat einen Index, auf dessen Wert der Anwender nicht zugreifen kann. Da für die Kalkulation ein Index erforderlich ist (in diesem Fall k), erstellen wir einen eigenen Index k, der innerhalb der Schleife, jedes Mal wenn das Programm die Schleife durchläuft, inkrementiert wird. Folgendes Programm wäre eine Möglichkeit, S zu berechnen:

```
« 0. DUP →n S k « 0. n START k SQ S + 1. 'k' STO+ 'S' STO
NEXT S "S" TAG » »
```

Speichern Sie das Programm unter S1.

Hier eine kurze Erklärung, wie das Programm arbeitet:

1. Das Programm benötigt als Eingabe eine Ganzzahl. Vor der Ausführung befindet sich diese Zahl (n) in Stack-Ebene 1. Das Programm wird gestartet.
2. Eine Null wird eingegeben und n in die Stack-Ebene 2 geschoben.
3. Der Befehl DUP, der über die Tastenfolge ALPHA ALPHA D U P ALPHA eingegeben werden kann, kopiert den Inhalt der Stack-Ebene 1, verschiebt alle Stack-Ebenen nach oben und schreibt die gerade erstellte Kopie in die Ebene 1. Nachdem DUP ausgeführt wurde, befindet sich n in der Stack-Ebene 3 und die Ebenen 1 und 2 enthalten Nullen.
4. Der Programmcode `→ n S k` speichert die Werte von n, 0 und 0 in die lokalen Variablen n, S, k. Man sagt, dass die Variablen n, S und k <u>initialisiert</u> wurden (S und k auf 0 und n auf den Wert, den der Anwender ausgewählt hat).
5. Der Programmcode `0. n START` beschreibt eine START-Schleife deren Index die Werte 0, 1, 2, …, n annehmen wird.
6. Die Summe S wird durch den folgenden Programmcode mit k^2 inkrementiert: `k SQ S +`
7. Der Index k wird durch den folgenden Programmcode mit 1 inkrementiert: `1. k +`
8. • An dieser Stelle stehen die aktuellen Werte für S und k in den Stack-Ebenen 2 und 1. Der Programmcode 'k' STO speichert den Wert der Stack-Ebene 1 in die lokale Variable k. Danach steht der aktuelle Wert von S in der Ebene 1.
9. • Der Programmcode `'S' STO` speichert den Wert der Stack-Ebene 1 in die lokale Variable k. Danach ist der Stack leer.
10. • Der Teil NEXT erhöht den Index um 1 und setzt das Programm an den Beginn der Schleife (Schritt 6).
11. • Die Schleife wird wiederholt, bis der Index den maximalen Wert n erreicht hat.
12. • Der letzte Teil des Programms ruft den letzten Wert von S (die Summation) auf, markiert ihn und schriebt ihn in die Stack-Ebene 1, wo

Sie als Ausgabe angezeigt wird.

Um das Programm Schritt für Schritt zu betrachten, verwenden Sie den Debugger wie folgt (n = 2). SL1 bedeutet Stack-Ebene 1:

VAR 2 ['] S1 ENTER	Schreibt Sie eine 2 in Ebene 2 und den Programmnamen ‚S1' in Ebene 1
← PRG NXT NXT RUN DBG	Start der Fehlersuche (des Debuggers) SL1 = 2.
SST↓	SL1 = 0., SL2 = 2.
SST↓	SL1 = 0., SL2 = 0., SL3 = 2. (DUP)
SST↓	Stack leeren(-> n S k)
SST↓	Stack leeren (« - Start des Unterprogramms)
SST↓	SL1 = 0., (Startwert des Schleifenindex)
SST↓	SL1 = 2.(n), SL2 = 0 (Endwert des Schleifen-) Index
SST↓	Stack leeren (START – Beginn der Schleife)

--- Loopdurchlauf 1 für k = 0

SST↓	SL1 = 0. (k)
SST↓	SL1 = 0. (SQ(k) = k^2)
SST↓	SL1 = 0.(S), SL2 = 0. (k^2)
SST↓	SL1 = 0. (S + k^2)
SST↓	SL1 = 1., SL2 = 0. (S + k^2)
SST↓	SL1 = 0.(k), SL2 = 1., SL3 = 0. (S + k^2)
SST↓	SL1 = 1.(k+1), SL2 = 0. (S + k^2)
SST↓	SL1 = 'k', SL2 = 1., SL3 = 0. (S + k^2)
SST↓	SL1 = 0. (S + k^2) [Speichert Wert von SL2 = 1, in SL1 = ‚k']
SST↓	SL1 = 'S', SL2 = 0. (S + k^2)
SST↓	Stack leeren [Speichert Wert von SL2 = 0, in SL1 = ‚S'.
SST↓	Stack leeren (NEXT – Ende der Schleife)

--- Loopdurchlauf 3 für k = 1

SST↓	SL1 = 1. (k)
SST↓	SL1 = 1. (SQ(k) = k^2)
SST↓	SL1 = 0.(S), SL2 = 1. (k^2)
SST↓	SL1 = 1. (S + k^2)
SST↓	SL1 = 1., SL2 = 1. (S + k^2)
SST↓	SL1 = 1.(k), SL2 = 1., SL3 = 1. (S + k^2)
SST↓	SL1 = 2.(k+1), SL2 = 1. (S + k^2)
SST↓	SL1 = 'k', SL2 = 2., SL3 = 1. (S + k^2)
SST↓	SL1 = 1. (S + k^2) [Speichert Wert von SL2 = 2, in SL1 = ‚k']
SST↓	SL1 = 'S', SL2 = 1. (S + k^2)
SST↓	Stack leeren [Speichert Wert von SL2 = 1, in SL1 = ‚S'.
SST↓	Stack leeren (NEXT – Ende der Schleife)

-- Loopdurchlauf 3 für k = 2

SST↓	SL1 = 2. (k)
SST↓	SL1 = 4. (SQ(k) = k^2)
SST↓	SL1 = 1.(S), SL2 = 4. (k^2)
SST↓	SL1 = 5. (S + k^2)
SST↓	SL1 = 1., SL2 = 5. (S + k^2)
SST↓	SL1 = 2.(k), SL2 = 1., SL3 = 5. (S + k^2)
SST↓	SL1 = 3.(k+1), SL2 = 5. (S + k^2)
SST↓	SL1 = 'k', SL2 = 3., SL3 = 5. (S + k^2)
SST↓	SL1 = 5. (S + k^2) [Speichert Wert von SL2 = 3, in SL1 = ‚k']
SST↓	SL1 = 'S', SL2 = 5. (S + k^2)
SST↓	Stack leeren [Speichert Wert von SL2 = 0, in SL1 = ‚S'.
SST↓	Stack leeren (NEXT – Ende der Schleife)

-- bei n = 2 ist der Schleifenindex verbraucht und das Programm fährt mit den Befehlen nach NEXT fort

SST↓	SL1 = 5 (S wird in den Stack geladen)
SST↓	SL1 = "S", SL2 = 5 ("S" wird in den Stack

	geschrieben)
SST↓	SL1 = S:5 (markieren des Ausgabewerts)
SST↓	SL1 = S:5 (Unterprogramm verlassen »)
SST↓	SL1 = S:5 (Hauptprogramm verlassen »)

Dies ist das Ende der Debugger-Liste Das Ergebnis des Programms S1 mit n = 2 beträgt S:5.

Prüfen Sie auch die folgenden Ergebnisse: VAR

3 S1	Ergebnis: `S:14`	4 S1	Ergebnis: `S:30`
5 S1	Ergebnis: `S:55`	8 S1	Ergebnis: `S:204`
10 S1	Ergebnis: `S:385`	20 S1	Ergebnis: `S:2870`
30 S1	Ergebnis: `S:9455`	100 S1	Ergebnis: `S:338350`

Die START…STEP Konstruktion

Die allgemeine Form des Befehls sieht wie folgt aus:

```
start_value end_value START program_statements increment
NEXT
```

start_value, end_value und `increment` des Schleifenindex können positive oder negative Werte haben. Bei einem `increment > 0` wird die Schleife so lange ausgeführt, wie der Index kleiner oder gleich `end_value` ist. Bei einem `increment < 0` wird die Schleife so lange ausgeführt, wie der Index größer oder gleich `end_value` ist.

Beispiel – Erzeugen einer Werteliste

Nehmen Sie an, das Sie eine Werteliste für x von x = 0.5 bis x = 6.5 in Schritten von 0.5 erzeugen wollen. Sie können das folgende Programm verwenden:

```
« → xs xe dx « xs DUP xe START DUP dx + dx STEP DROP xe
xs – dx / ABS 1 + →LIST » »
```

und es in die Variable GLIST speichern.

In diesem Programm sind xs = Startwert der Schleife, xe = Endwert der Schleife und dx = Inkrement der Schleife. Das Programm schreibt Werte für xs, xs+dx, xs+2·dx, xs+3·dx, … in den Stack. Dann errechnet es mit dem folgenden Programmcode die Anzahl der erzeugten Elemente:

```
xe xs – dx / ABS 1. +
```

Am Ende erstellt das Programm eine Liste mit den Elementen aus dem Stack.

- Prüfen Sie, ob der Programmaufruf 0.5 ENTER 2.5 ENTER 0.5 ENTER GLIST folgende Liste {0.5 1. 1.5 2. 2.5} generiert.
- Um die Ausführung Schritt für Schritt zu betrachten, verwenden Sie das Programm DBUG mit einer kurzen Liste, beispielsweise:

VAR 1 SPC 1.5 SPC 0.5 ENTER	Parameter 1 1.5 0.5 eingeben
['] GLIST ENTER	Programmnamen in Ebene 1
← PRG NXT NXT RUN DBG	Start der Fehlersuche (des Debuggers)

Verwenden Sie SST↓ , um in das Programm zu springen und zu beobachten, wie die einzelnen Befehle arbeiten.

Der Befehl FOR

Wie bei dem Befehl START, gibt es auch bei FOR zwei Varianten: Die FOR…NEXT Konstruktion, bei einer Inkrementierung des Index um 1 und die FOR…STEP Konstruktion mit einer Inkrementierung des Index, die vom Anwender gewählt werden kann. Anders als beim Befehl START müssen wir beim Befehl FOR für den Schleifenindex einen Namen vergeben (z.B. j, k, n). Wir müssen uns aber nicht wie bei START um die Inkrementierung des Index selbst kümmern. Der entsprechende Wert des Index kann für Berechnung verwendet werden.

Die Befehle der FOR-Konstruktion rufen Sie wie folgt auf:

← PRG BRCH FOR

Innerhalb des Menüs BRCH (PRG BRCH) stehen folgende Tastenfolgen zur Verfügung, um eine FOR-Konstruktion zu erstellen (das Symbol ⇐ zeigt die Cursor-Position):

- FOR: Startet die FOR…NEXT Konstruktion: FOR ⇐ NEXT
- FOR: Startet die FOR…STEP Konstruktion: FOR ⇐ STEP

Die FOR…NEXT Konstruktion

Die allgemeine Form des Befehls sieht wie folgt aus:

```
start_value end_value FOR loop_index program_statements
NEXT
```

Um eine Endlosschleife zu verhindern, stellen Sie sicher, dass `start_value < end_value` ist.

Beispiel – Berechnung der Summation S mit einer FOR…NEXT Konstruktion
Das folgende Programm berechnet die Summation

$$S = \sum_{k=0}^{n} k^2$$

Verwenden Sie eine FOR…NEXT Schleife:

```
« 0 →n S « 0 n FOR k k SQ S + 'S' STO NEXT S "S" TAG » »
```

Speichern Sie dieses neue Programm in die Variable S1. Versuchen Sie folgendes Beispiel: VAR

3 S1	Ergebnis: S:14	4 S1	Ergebnis: S:30
5 S1	Ergebnis: S:55	8 S1	Ergebnis: S:204
10 S1	Ergebnis: S:385	20 S1	Ergebnis: S:2870
30 S1	Ergebnis: S:9455	100 S1	Ergebnis: S:338350

Sie werden bemerkt haben, dass dieses Programm sehr viel einfacher ist, als das unter
S1 gespeicherte. Der Index k muss innerhalb des Programms weder initialisiert noch inkrementiert werden. Das Programm selbst übernimmt diese Aufgaben.

Die FOR…STEP Konstruktion

Die allgemeine Form des Befehls sieht wie folgt aus:

```
start_value end_value FOR loop_index program_statements
increment STEP
```

start_value, end_value und `increment` des Schleifenindex können positive oder negative Werte haben. Bei einem `increment > 0` wird die Schleife so lange ausgeführt, wie der Index kleiner oder gleich `end_value` ist. Bei einem `increment < 0` wird die Schleife so lange ausgeführt, wie der Index größer oder gleich `end_value` ist. Die Programmbefehle werden zumindest ein Mal ausgeführt (z.B. `1 0 START 1 1 STEP` gibt 1 zurück).

Beispiel – Erzeugen einer Werteliste mit einer FOR…STEP Konstruktion
Geben Sie das folgende Programm ein

```
« → xs xe dx « xe xs – dx / ABS 1. + → n « xs xe FOR x
x dx STEP n →LIST » » »
```

und speichern es in die Variable GLIS2 .

- Prüfen Sie, ob der Programmaufruf 0.5 ENTER 2.5 ENTER 0.5 ENTER GLIS2 folgende Liste {0.5 1. 1.5 2. 2.5} generiert.
- Um die Ausführung Schritt für Schritt zu betrachten, verwenden Sie das Programm DBUG mit einer kurzen Liste, beispielsweise:

VAR 1 SPC 1.5 SPC 0.5 ENTER	Parameter 1 1.5 0.5 eingeben
['] GLIS2 ENTER	Programmnamen in Ebene 1
← PRG NXT NXT RUN DBG	Start der Fehlersuche (de Debuggers)

Verwenden Sie SST↓ , um in das Programm zu springen und zu beobachten, wie die einzelnen Befehle arbeiten.

Der Befehl DO

Die allgemeine Form des Befehls sieht wie folgt aus:

```
DO program_statements UNTIL logical_statement END
```

Der DO Befehl startet eine unendliche Schleife und führt die program_statements aus, bis logical_statement unwahr (0) zurückgibt. `logical_statement` muss den Wert eines Index enthalten, dessen Wert durch `program_statements` verändert wird.

<u>Beispiel 1</u> – Dieses Programm erzeugt in der oberen linken Ecke der Anzeige einen Zähler, der in einer Endlosschleife 1 aufaddiert, bis der Druck auf eine Taste den Zähler stoppt: `« 0 DO DUP 1 DISP 1 + UNTIL KEY END DROP »`

Der Befehl KEY erkennt TRUE, wenn eine Taste gedrückt wird.

<u>Beispiel 2</u> – Berechnung der Summation S mit einer DO...UNTIL...END Konstruktion
Das folgende Programm berechnet die Summation

$$S = \sum_{k=0}^{n} k^2$$

Verwenden Sie eine DO…UNTIL…END Schleife:

```
« 0. →n S « DO n SQ S + ‘S‘ STO n 1 – ‘n‘ STO UNTIL
‘n<0‘ END S “S” TAG » »
```

Speichern Sie neue Programm in die Variable S3. Versuchen Sie folgendes Beispiel: VAR

3	S3	Ergebnis: `S:14`	4	S3	Ergebnis: `S:30`
5	S3	Ergebnis: `S:55`	8	S3	Ergebnis: `S:204`
10	S3	Ergebnis: `S:385`	20	S3	Ergebnis: `S:2870`

30 [S2]   Ergebnis: `S:9455`   100 [S2]   Ergebnis: `S:338350`

<u>Beispiel 3</u> – Erzeugen einer Liste mit einer DO…UNTIL…END Konstruktion
Geben Sie das folgende Programm ein

```
« → xs xe dx « xe xs – dx / ABS 1. + xs → n x « xs DO
'x+dx' EVAL DUP 'x' STO UNTIL 'x≥xe' END n →LIST » » »
```

und speichern es in die Variable [GLIS3] .

- Prüfen Sie, ob der Programmaufruf 0.5 [ENTER] 2.5 [ENTER] 0.5 [ENTER] [GLIS3] folgende Liste {0.5 1. 1.5 2. 2.5} generiert.
- Um die Ausführung Schritt für Schritt zu betrachten, verwenden Sie das Programm DBUG mit einer kurzen Liste, beispielsweise:

[VAR] 1 [SPC] 1.5 [SPC] 0.5 [ENTER]	Parameter 1 1.5 0.5 eingeben
['] [GLIS3] [ENTER]	Programmnamen in Ebene 1
[←] PRG [NXT] [NXT] [RUN] [DBG]	Start der Fehlersuche (des Debuggers)

Verwenden Sie [SST↓] , um in das Programm zu springen und zu beobachten, wie die einzelnen Befehle arbeiten.

Der Befehl WHILE

Die allgemeine Form des Befehls sieht wie folgt aus:

```
WHILE logical_statement REPEAT program_statements END
```

Der Befehl WHILE wiederholt `program_statements` so lange, wie `logical_statement` wahr ist (nicht Null). Ist dies nicht mehr der Fall, fährt das Programm mit den Befehlen direkt nach END fort. `program_statements` müssen einen Index enthalten, der geändert wird, bevor `logical_statement` zu Beginn der nächsten Wiederholung überprüft wird. Anders als beim Befehl DO wird die Schleife, wenn die Überprüfung von logical_statement beim ersten Mal unwahr zurück gibt, nie ausgeführt.

Beispiel 1 – Berechnung der Summation S mit einer WHILE…REPEAT…END Konstruktion
Das folgende Programm berechnet die Summation

$$S = \sum_{k=0}^{n} k^2$$

Verwenden einer WHILE…REPEAT…END Schleife:

```
« 0. →n S « WHILE 'n≥0' REPEAT n SQ S + 'S' STO n 1 –
'n' STO END S "S" TAG » »
```

Speichern Sie neue Programm in die Variable [S4]. Versuchen Sie folgendes Beispiel: [VAR]

3	[S4]	Ergebnis: S:14	4	[S4]	Ergebnis: S:30
5	[S4]	Ergebnis: S:55	8	[S4]	Ergebnis: S:204
10	[S4]	Ergebnis: S:385	20	[S4]	Ergebnis: S:2870
30	[S4]	Ergebnis: S:9455	100	[S4]	Ergebnis: S:338350

Beispiel 2 – Erzeugen einer Liste mit einer WHILE…REPEAT...END Konstruktion
Geben Sie das folgende Programm ein:

```
« → xs xe dx « xe xs – dx / ABS 1. + xs → n x « xs
WHILE 'x<xe'REPEAT 'x+dx' EVAL DUP 'x' STO END n →LIST » »
»
```

und speichern es in die Variable [GLIS4] .

- Prüfen Sie, ob der Programmaufruf 0.5 [ENTER] 2.5 [ENTER] 0.5 [ENTER] [GLIS4] folgende Liste {0.5 1. 1.5 2. 2.5} generiert.
- Um die Ausführung Schritt für Schritt zu betrachten, verwenden Sie das Programm DBUG mit einer kurzen Liste, beispielsweise:

[VAR] 1 [SPC] 1.5 [SPC] 0.5 [ENTER]	Parameter 1 1.5 0.5 eingeben
['] [illegible] [ENTER]	Programmnamen in Ebene 1
[←] PRG [NXT] [NXT] [RUN] [DBG]	Start der Fehlersuche (des Debuggers)

Verwenden Sie [SST↓] , um in das Programm zu springen und zu beobachten, wie die einzelnen Befehle arbeiten.

Fehler und Fehler abfangen

Mit Hilfe der Funktionen des Menüs PRG/ERROR können Sie Fehler im Rechner und Fallen in Programmen manipulieren. Das Menü PRG/ERROR können Sie mit der Tastenfolge [←] PRG [NXT] [NXT] [ERROR] aufrufen. Es enthält die folgenden Funktionen und Untermenüs:

DOERR

Diese Funktion simuliert einen benutzerdefinierten Fehler, wobei sich der Rechner so verhält, als wäre dieser Fehler tatsächlich aufgetreten. Der Funktion kann als Argument eine Ganzzahl, eine binäre Ganzzahl, eine Fehlermeldung oder eine Null (0) zugeordnet werden. Geben Sie beispielsweise im RPN-Modus [5] [ENTER] [DOERR], erscheint die folgende Fehlermeldung: *Error: Memory Clear*

Wenn Sie #11h [ENTER] [DOERR] eingeben, erscheint die folgende Meldung: *Error: Undefined FPTR Name*

Wenn Sie "TRY AGAIN" [ENTER] [DOERR] eingeben, erscheint die folgende Meldung: *TRY AGAIN*

[0] [ENTER] [DOERR], erzeugt die folgende Meldung: *Interrupted*

ERRN

Diese Funktion gibt die Kennziffer des letzten Fehlers zurück. Wenn Sie beispielsweise 0 1/x ON ERRN versuchen, erscheint die Nummer #305h. Diese binäre Ganzzahl steht für den Fehler *Infinite Result*

ERRM

Diese Funktion gibt eine Zeichenkette mit der Fehlermeldung des letzten Fehlers zurück. Wenn Sie beispielsweise 0 1/x ON ERRM versuchen, erscheint die folgende Zeichenkette: *"Infinite Result"*

ERR0

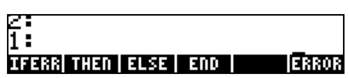
Mit dieser Funktion wird die letzte Fehlernummer gelöscht, so dass die Funktion ERRN #0h zurückgibt. Wenn Sie beispielsweise 0 1/x ON ERR0 ERRN versuchen, erhalten Sie # 0h. Wenn Sie 0 1/x ON ERR0 ERRM versuchen, erhalten Sie eine leere Zeichenkette " ".

LASTARG

Diese Funktion gibt Kopien der Argumente des zuletzt ausgeführten Befehls oder Funktion zurück. Wenn Sie im RPN-Modus 3 ÷ 2 ENTER eingeben und dann LASTARG (LASTA) benutzen, werden im Stack die Werte 3 und 2 aufgelistet. Wenn Sie im RPN-Modus 5 TAN ENTER eingeben, erzeugt LASTARG eine 5.

Untermenü IFERR

Das Untermenü IFERR bietet die folgenden Funktionen:



Dies sind die Befehle der IFERR … THEN … END oder IFERR … THEN … ELSE … END Konstruktionen. Mit beiden logischen Konstruktionen können Fehler, die bei der Programmausführung auftreten, abgefangen werden. Wenn Sie im Untermenü ERROR ← IFERR oder → IFERR eingeben, werden die IFERR-Befehle in den Stack geschrieben und können dort vervollständigt werden:

Die allgemeine Form der beiden Konstruktionen sieht wie folgt aus:

IF Abfangbedingung THEN Fehlerbedingung END

IF Abfangbedingung THEN Fehlerbedingung ELSE Normalbedingung END

Die Arbeitsweise dieser logischen Konstruktionen entspricht denen, die Sie bei IF … THEN … END und IF … THEN … ELSE … END bereits kennen gelernt haben. Wird während der Ausführung der Abfangbedingungen ein Fehler entdeckt, wird die Fehlerbedingung ausgeführt. In allen anderen Fällen werden die Normalbedingungen ausgeführt.

Betrachten Sie beispielsweise das folgende Programm (ERR1), das als Eingabe zwei Matrizen A und b verwendet und dabei die Abfangbedingung auf Fehler prüft: A b / (RPN-Modus, also A/b). Tritt ein Fehler auf, ruft das Programm die Funktion LSQ (Least SQuares, siehe Kapitel 11) auf, um das Gleichungssystem zu lösen:

```
« → A b « IFERR A b / THEN LSQ END »  »
```

Testen Sie das Programm mir den Argumenten A = [[2, 3, 5] , [1, 2, 1]] und b = [[5] , [6]]. Eine einfache Division dieser Argumente verursacht einen Fehler: */Error: Invalid Dimension.*

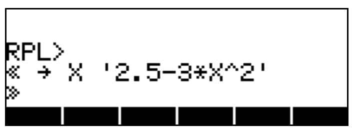
Mit der Fehlerabfrage des Programms ERR1 jedoch ergibt sich mit den gleichen Argumenten : [0.262295…, 0.442622…].

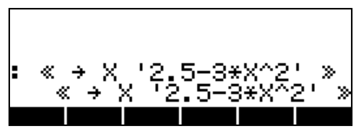
RPL-Programmierung im algebraischen Modus

Obwohl alle Programme, die wir bisher vorgestellt haben, im RPN-Modus erstellt und auch ausgeführt wurden, können Sie, wenn Sie sich im algebraischen Modus befinden, Programme auch mit Hilfe der Funktion RPL>

in RPL eingeben. Die Funktion wird über den Befehlskatalog aufgerufen. Als Beispiel, versuchen Sie dasselbe Programm im ALG-Modus und speichern Sie dies in der Variablen P2: « → X '2.5-3*X^2' »

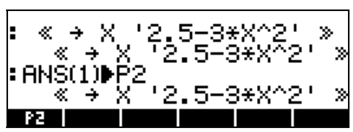
Starten Sie zuerst die RPL> Funktion aus dem Befehle Katalog (→ _CAT_). Alle im ALG-Modus gestarteten Funktionen werden, anhand ihres Namens, in Klammern dargestellt. Die Funktion RPL> ist keine Ausnahme, Sie müssen aber, bevor Sie ein Programm in den Display eingeben, die Klammern entfernen. Zum Löschen der Klammern aus der RPL>() Anweisung verwenden Sie die Pfeiltasten (◀ ▶) sowie die Löschtaste (◀). An dieser Stelle können Sie das RPL Programm eingeben. Nachfolgende Abbildung zeigt den RPL> Befehl zusammen mit dem Programm vor und nach drücken der Taste ENTER.





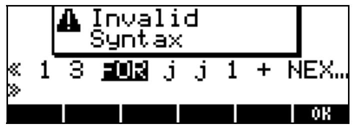
Benutzen Sie den STO Befehl, um dieses Programm zu speichern.

← ANS STO▶ ALPHA P 2 ENTER

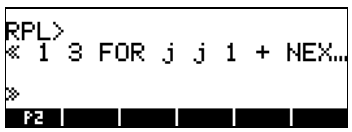


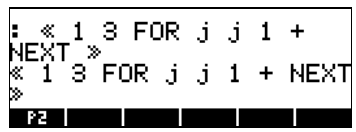
Eine Auswertung des Programms P2 für das Argument X = 5 wird in der nachfolgenden Abbildung dargestellt:

Während Sie im ALG-Modus das Programm ohne die RPL> Funktion erstellen können, erzeugen einige RPL Zusammensetzungen, sobald Sie die Taste ENTER drücken, eine Fehlermeldung, so z.B.:



Dagegen, wenn Sie RPL verwenden, gibt es keine Probleme, wenn Sie das Programm im ALG-Modus laden:





Kapitel 22
Programme für die Manipulation von Grafiken

Dieses Kapitel enthält einige Beispiele, mit denen die Funktionen des Rechners für die interaktive bzw. programmgesteuerte Manipulation von Grafiken erläutert werden. Wie in Kapitel 21, empfiehlt sich auch hier die Verwendung des RPN-Modus, sowie das Setzen des System-Flags 117 auf SOFT-Menüeinträge. « »

In Kapitel 12 wurden mehrere Grafikanwendungen des Rechners vorgeführt. Bei den Beispielen in Kapitel 12 ging es um die interaktive Erzeugung von Grafiken mit Hilfe der vorprogrammierten Eingabemasken des Rechners. Grafiken können auch in Ihren Programmen eingesetzt werden, z.B. für die Erläuterung der numerischen Ergebnisse. Für die Lösung solcher Aufgaben werden zuerst die Funktionen im PLOT-Menü beschrieben.

Das Menü PLOT

Befehle zum Einstellen und Erstellen von Plots finden Sie im Menü PLOT. Ins PLOT Menü gelangen Sie mit nachfolgender Tastenfolge: 8 1 . 0 1 ← PRG NXT MODES MENU MENU.

Das so erzeugte Menü erlaubt dem Anwender Zugang zu einer Vielzahl von Grafik-Funktionen. Definieren wir die Taste F3 (GRAPH) für einen einfachen Zugang zum beschriebenen Menü, für die nachfolgenden Beispiele:

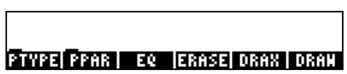
Benutzerdefinierte Taste für das PLOT-Menü

Mit der folgenden Tastenfolge überprüfen Sie, ob auf dem Rechner bereits benutzerdefinierte Tasten gespeichert wurden.
← PRG NXT MODES KEYS RCLK.

Falls keine benutzerdefinierte Tasten gespeichert sind, wird eine Liste mit einem S zurückgegeben (z.B. {S}). Dies bedeutet, dass auf Ihrem Rechner nur die standardmäßige Tastendefinition gespeichert ist.
Um eine Taste als benutzerdefinierte Taste anzugeben, müssen Sie zu dieser Liste einen Befehl bzw. ein Programm hinzufügen, gefolgt von der Angabe der Taste (ausführliche Details finden Sie in Kapitel 20). Geben Sie die nachfolgende Liste
`{ 8 << 81.01 MENU >> 13.0 }` in den Stack ein und verwenden Sie die Funktion STOREKEYS (← PRG NXT MODES KEYS STOK), um die Taste F3 als Zugriffstaste für das PLOT-Menü zu definieren. Mit ← PRG NXT MODES KEYS RCLK überprüfen Sie, ob die Liste auf dem Rechner gespeichert wurde.

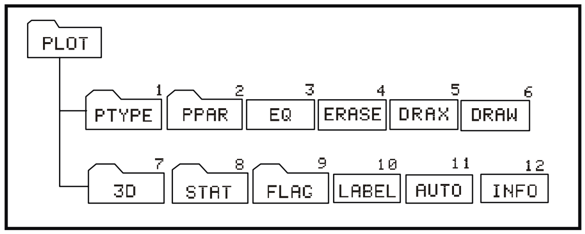
Anmerkung: Bei der Beschreibung des PLOT-Menüs, sowie seiner Funktionen oder Untermenüs, werden keine Beispiele aufgeführt. In diesem Abschnitt werden wir uns eher auf die Inhalte des PLOT-Menüs, bezogen auf verschiedenen Grafikarten, konzentrieren.

Um eine benutzerdefinierte Taste zu aktivieren, müssen Sie zuerst ← USER drücken (genau wie bei der Taste ALPHA), bevor Sie die gewünschte Taste oder Tastenfolge drücken. Um das PLOT-Menü mit der oben angegebenen Tastendefinition zu starten, drücken Sie ← USER F3. Das folgende Menü erscheint (drücken Sie NXT, um sich zum zweiten Menü zu bewegen).



Beschreibung des Menüs PLOT

Die folgende Abbildung zeigt die Einträge des PLOT-Menüs. Die Zahlen oberhalb der Menüs und Funktionen dienen weiterhin als Referenz in der nachfolgenden Beschreibung dieser Objekte.



Die Funktionstasten 3D, STAT, FLAG, PTYPE und PPAR haben auch eigene Untermenüs, die zu einem späteren Zeitpunkt ausführlicher beschrieben werden. Jetzt werden nur die Funktionen beschrieben, die im Menü Nummer 81.02 direkt mit den Funktionstasten aufgerufen werden können. Diese Funktionen lauten:

LABEL (10)
Mit der Funktion LABEL werden die Achsen eines Plots beschriftet, einschließlich der Variablennamen bzw. der minimalen und maximalen Werte der Achsen. Die Namen der Variablen werden aus den Informationen in der Variable PPAR zusammengestellt.

AUTO (11)
Die Funktion AUTO (AUTO Maßstab) berechnet den Anzeigebereich für die y-Achse oder für beide Achsen (x und y) in zweidimensionalen Plots, je nach dem Plottyp, der in PPAR definiert wurde. Bei dreidimensionalen Grafiken bewirkt die Funktion AUTO nichts. Bei zweidimensionalen Plots werden von der Funktion AUTO die folgenden Aktionen ausgeführt:

- FUNCTION : Je nach Plotbereich von x, nimmt die Funktion eine Stichprobe in EQ; anschließend werden die Minima und Maxima von y bestimmt.
- CONIC : Der Maßstab der y-Achse wird mit dem der x-Achse gleich gesetzt.
- POLAR : Basierend auf den Werten der unabhängigen Variablen (meistens θ), nimmt die Funktion eine Stichprobe in EQ; anschließend werden die Minima und Maxima von x und y bestimmt.

- PARAMETRIC : Führt zu einem ähnlichen Ergebnis wie POLAR, bezogen auf die Werte des Parameters, der die Gleichungen für x und y definiert.
- TRUTH : Hat keine Wirkung.
- BAR : Der Bereich der x-Achse wird zwischen 0 und *n+1* gesetzt, wobei *n* die Anzahl der Elemente aus ΣDAT ist. Der Wertebereich von y hängt vom Inhalt von ΣDAT ab. Die minimalen und maximalen Werte von y werden so bestimmt, dass die x-Achse immer in der Grafik eingeschlossen ist.
- HISTOGRAM : Ähnlich wie BAR.
- SCATTER : Setzt die Bereiche der x- und y-Achsen, abhängig vom Inhalt der unabhängigen und abhängigen Variablen aus ΣDAT.

INFO (12)
Die Funktion INFO ist ausschließlich interaktiv (d.h. sie ist nicht programmierbar). Wird die entsprechende Funktionstaste gedrückt, so werden Informationen über die aktuellen Plotparameter angezeigt.

EQ (3)
Der Variablenname EQ ist für die Speicherung der aktuellen Gleichung in Plots bzw. der Lösungen von Gleichungen vorbehalten (siehe Kapitel ...). Die Funktionstaste EQ aus diesem Menü kann so verwendet werden, als wäre das Variablenmenü verfügbar. Wenn z.B. [EQ] gedrückt wird, werden die aktuelle Inhalte der Variable angezeigt.

ERASE (4)
Die Funktion ERASE löscht den aktuellen Inhalt des Grafikfensters. Während der Programmierung wird mit dieser Funktion sichergestellt, dass das Grafikfenster vor dem Plotten einer neuen Grafik gelöscht wird.

DRAX (5)
Die Funktion DRAX zeichnet die Achsen im aktuellen Plot, falls eine sichtbar ist.

DRAW (6)

Die Funktion DRAW zeichnet den Plot, der in PPAR definiert wurde.

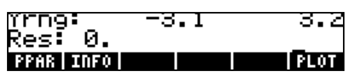
Das Menü PTYPE unter PLOT (1)

Das Menü PTYPE listet die Namen aller zweidimensionalen, im Rechner vorprogrammierten Plottypen auf. Das Menü enthält die folgenden Funktionstasten:

Diese Tasten entsprechen den Plottypen *Function, Conic, Polar, Parametric, Truth* und *Diff Eq*, die oben beschrieben wurden. Wird eine dieser Funktionstasten gedrückt, während ein Programm eingegeben wird, so wird an dieser Stelle im Programm der entsprechende Funktionsaufruf eingefügt. Drücken Sie NXT PLOT , um zum Hauptmenü PLOT zurückzukehren.

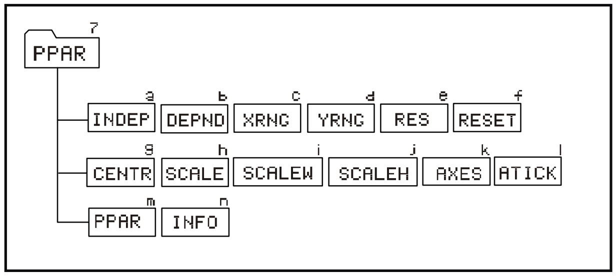
Das Menü PPAR (2)

Im PPAR-Menü stehen verschiedene Optionen für die PPAR-Variable zur Verfügung und diese entsprechen den folgenden Funktionstasten. Drücken Sie NXT , um sich zum nächsten Menü zu bewegen.

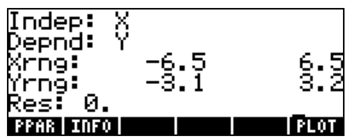


Anmerkung: Die hier dargestellten SCALE-Befehle sind eigentlich SCALE, SCALEW und SCALEH, uns zwar in dieser Reihenfolge.

Die folgende Abbildung zeigt die Funktionen aus dem PPAR-Menü. Die Buchstaben oberhalb der Funktionen dienen als Referenz in der nachfolgenden Beschreibung der Funktionen.



INFO (n) und PPAR (m)
Wenn Sie INFO drücken oder ↱ PPAR eingeben, während Sie sich in diesem Menü befinden, erscheint eine Liste mit den aktuellen PPAR-Einstellungen. Beispiel:



Diese Informationen haben die folgende Bedeutung: X ist eine unabhängige Variable (Indep), Y ist die abhängige Variable (Depnd), der Bereich der x-Achse liegt zwischen -6.5 und 6.5 (Xrng), der Bereich der y-Achse liegt zwischen -3.1 und 3.2 (Yrng). Die letzte Information auf dem Bildschirm, der Wert von Res (Auflösung), bestimmt das Intervall der unabhängigen Variablen, die für die Erstellung des Plots verwendet wird.

Die Funktionstasten im Menü PPAR(2) sind Befehle, die in Programmen eingesetzt werden können. Diese Befehle lauten:

INDEP (a)
Der Befehl INDEP gibt die unabhängige Variable und deren Plot-Bereich an. Diese Angaben werden in der Variable PPAR als dritter Parameter gespeichert. Der Vorgabewert ist 'X'. Die folgenden Werte können der unabhängigen Variable zugeordnet werden:

- Der Name einer Variable, z.B. '`Vel`'
- Der Name einer Variable aus einer Liste, z.B. { `Vel` }

- Der Name einer Variable und ein Bereich aus einer Liste, z.B. { `Vel 0 20` }
- Ein Bereich ohne einen Variablen-Namen, z.B. { `0 20` }
- Zwei Werte, die einen Bereich darstellen, z.B. `0 20`

In einem Programm steht nach all diesen Angaben der Befehl INDEP.

<u>DEPND (b)</u>
Der Befehl DEPND gibt den Namen der abhängigen Variable an. Bei TRUTH-Plots spezifiziert er außerdem den Plot-Bereich. Der Vorgabewert ist Y. Die Angaben für die DEPND-Variable sind identisch mit den Angaben für die INDEP-Variable.

<u>XRNG (c) und YRNG (d)</u>
Der Befehl XRNG gibt den Plotbereich für die x-Achse an, und der Befehl YRNG den Plot-Bereich für die y-Achse. Diese Befehle erfordern die Eingabe der Minima bzw. Maxima von x bzw. y. Die Werte für die Achsenbereiche von x und y werden in den ersten zwei Elementen der Variable PPAR als geordnete Paare (x_{min}, y_{min}) und (x_{max}, y_{max}) gespeichert. Die Vorgabewerte für x_{min} und x_{max} sind jeweils -6,5 und 6,5. Die Vorgabewerte für x_{min} und x_{max} sind jeweils -3.1 und 3.2.

<u>RES (e)</u>
Der Befehl RES (RESolution - Auflösung) gibt das Intervall zwischen den Werten der unabhängigen Variable an, wenn ein bestimmter Plot erstellt wird. Die Auflösung kann entweder in Benutzereinheiten als eine Realzahl oder in Pixel als binäre Ganzzahl (Zahlen, die mit # anfangen, wie z.B. #10) angegeben werden. Die Auflösung wird in der Variable PPAR als vierter Parameter gespeichert.

<u>CENTR (g)</u>
Der Befehl CENTR hat als Argument ein geordnetes Paar (x,y) oder einen Wert x, und passt die ersten zwei Elemente in der Variable PPAR (z.B. (x_{min}, y_{min}) und (x_{max}, y_{max})) so an, dass die Mitte des Plots sich in (x,y) bzw. (x,0) befindet.

SCALE (h)

Der Befehl SCALE bestimmt den Plot-Maßstab, der als Anzahl von Benutzereinheiten pro Tick-Zeichen angegeben wird. Der Vorgabewert beträgt 1 Benutzereinheit pro Tick-Zeichen. Der Befehl SCALE benötigt zwei Zahlen als Argumente, x_{scale} und y_{scale} für die neuen horizontalen und vertikalen Maßstäbe. Dieser Befehl bewirkt, dass die Parameter (x_{min}, y_{min}) und (x_{max}, y_{max}) in PPAR an den gewünschten Maßstab angepasst werden. Die Mitte des Plots bleibt erhalten.

SCALEW (i)

Wird ein Faktor x_{factor} angegeben, multipliziert der Befehl SCALEW den horizontalen Maßstab mit diesem Faktor. Das W aus SCALEW steht für das englische Wort für Breite (width). Nach der Ausführung des SCALEW-Befehls werden die Werte x_{min} und x_{max} in PPAR geändert.

SCALEH (j)

Wird ein Faktor y_{factor} angegeben, multipliziert der Befehl SCALEH den vertikalen Maßstab mit diesem Faktor. Das H aus SCALEH steht für das englische Wort für Höhe (height). Nach der Ausführung des SCALEH-Befehls werden die Werte y_{min} and y_{max} in PPAR geändert.

Anmerkung: Die von SCALE, SCALEW oder SCALEH vorgenommenen Änderungen können zum Vergrößern oder Verkleinern eines Plots verwendet werden.

ATICK (l)

Der Befehl ATICK (TICK-Zeichen für Achsen) wird dazu verwendet, Tick-Zeichenvermerke für die Achsen vorzunehmen. Der Befehl ATICK akzeptiert die folgenden Eingabewerte:

- Ein Realwert x: setzt die Tick-Vermerke für beide Achsen (x und y) auf x-Einheiten
- Eine Liste mit zwei Realwerten { x y }: setzt die Tick-Vermerke für beide Achsen (x und y) auf x- bzw. y-Einheiten
- Eine binäre Ganzzahl #n: setzt die Tick-Vermerke für beide Achsen (x und y) auf #n Pixel

Eine Liste mit zwei binären Ganzzahlen { #n #m }: setzt die Tick-Vermerke für beide Achsen (x und y) auf #n bzw. #m Pixel.

AXES (k)
Der Eingabewert für den Befehl AXES besteht entweder aus einem geordneten Paar (x,y) oder aus einer Liste {(x,y) *atick* "Beschriftung x-Achse" "Beschriftung y-Achse"}. Der Parameter *atick* wird für die Angabe der Tick-Vermerke verwendet, wie bereits beim Befehl ATICKbeschrieben wurde. Das geordnete Paar ist die Mitte des Plots. Wird für AXES nur ein geordnetes Paar angegeben, so wird nur der Ursprung der Achsen geändert. Das Argument im Befehl AXES, egal ob geordnetes Paar oder Werteliste, wird als fünfter Parameter in PPAR gespeichert.

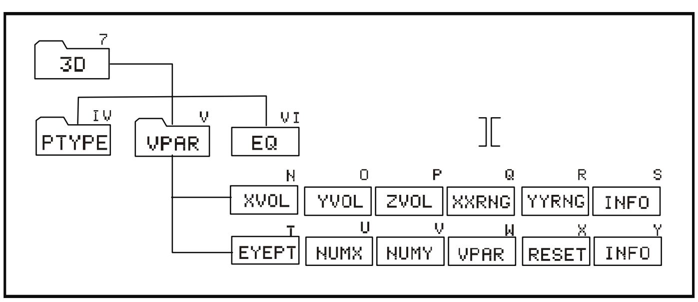
Drücken Sie PLOT, um zum PLOT Menü zurückzukehren.

Drücken Sie NXT, um zum zweiten Menü der Menüreihe PLOT zu gelangen.

RESET (f)
Diese Taste setzt die Plot-Parameter auf ihre Vorgabewerte zurück.

Das Menü 3D unter PLOT (7)

Das Menü 3D enthält zwei Untermenüs, PTYPE und VPAR, sowie eine Variable, EQ. Da die Bedeutung von EQ bereits erläutert wurde, werden wir uns hier auf die Inhalte der Menüs PTYPE und VPAR konzentrieren. Die nachstehende Darstellung zeigt die Struktur des 3D-Menüs.



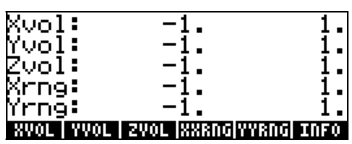
Das Menü PTYPE unter 3D (IV)

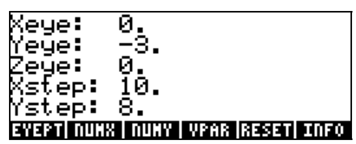
Das Menü PTYPE unter 3D enthält die folgenden Funktionen:

Diese Funktionen entsprechen den Grafikoptionen *Slopefield, Wireframe, Y-Slice, Ps-Contour, Gridmap* und *Pr-Surface*, die bereits in diesem Kapitel beschrieben wurden. Wird eine dieser Funktionstasten gedrückt, während ein Programm eingegeben wird, so wird an dieser Stelle im Programm der entsprechende Funktionsaufruf eingefügt. Drücken Sie NXT 3D , um zum Hauptmenü 3D zurückzukehren.

Das Menü VPAR unter 3D (V)

Die Variable VPAR steht für Volume PARameter (Volumenparameter) und sie bezieht sich auf einem Parallelepipeden im Raum, in dessen Innerem eine beliebige dreidimensionale Grafik erzeugt wird. Wenn im 3D-Menü [VPAR] gedrückt wird, werden die folgenden Funktionen aufgerufen. Drücken Sie NXT , um sich zum nächsten Menü zu bewegen.





Als nächstes wird die Bedeutung dieser Funktionen erläutert:

INFO (S) und VPAR (W)

Wird INFO (S) gedrückt, erscheinen die Informationen, die in der oberen Abbildung auf der linken Seite zu sehen sind. Die Bereiche *Xvol*, *Yvol* und *Zvol* bestimmen die Größe des Parallelepipeds, in dem die Grafik eingefügt wird. *Xrng* und *Yrng* sind die Wertebereiche von x bzw. y als unabhängige Variablen in der Ebene x-y, die für die Erzeugung von Funktionen der Art z = f(x,y) verwendet werden.

Drücken Sie NXT und INFO (Y), um die Informationen zu erhalten, die in der oberen Abbildung auf der rechte Seite dargestellt sind. Dieses sind die Werte für die Position des Blickwinkels für die dreidimensionale Grafik (Xeye, Yeye, Zeye), sowie die Anzahl der Schritte in x und y, die für die Erstellung eines Rasters zur Darstellung der Oberfläche erforderlich sind.

XVOL (N), YVOL (O) und ZVOL (P)
Diese Funktion erfordert die Eingabe des minimalen bzw. maximalen Wertes; die Funktion bestimmt die Größe des Parallelepipeds, in dem die Grafik erzeugt wird. Diese Werte werden in der Variable VPAR gespeichert. Die Vorgabewerte für die Bereiche XVOL, YVOL und ZVOL sind –1 bis 1.

XXRNG (Q) und YYRNG (R)
Diese Funktionen erfordern die Eingabe des minimalen bzw. maximalen Wertes; die Funktion bestimmt den Bereich der Variablen x und y, die für die Erstellung von Funktionen der Art z = f(x,y) verwendet werden. Die Vorgabewerte der Bereiche XXRNG und YYRNG sind die Gleichen wie bei XVOL und YVOL.

EYEPT (T)
Die Funktion EYEPT erfordert als Eingabe die Realwerte x, y und z als Position des Blickwinkels der dreidimensionalen Grafik. Der Blickwinkel ist ein Punkt im Raum von dem aus die dreidimensionale Grafik beobachtet wird. Wird der Blickwinkel geändert, so werden unterschiedliche Ansichten der Grafik dargestellt. Die Abbildung unten zeigt den Blickwinkel im Bezug auf den eigentlichen Grafikraum und seiner Projektion in der Anzeigenebene.

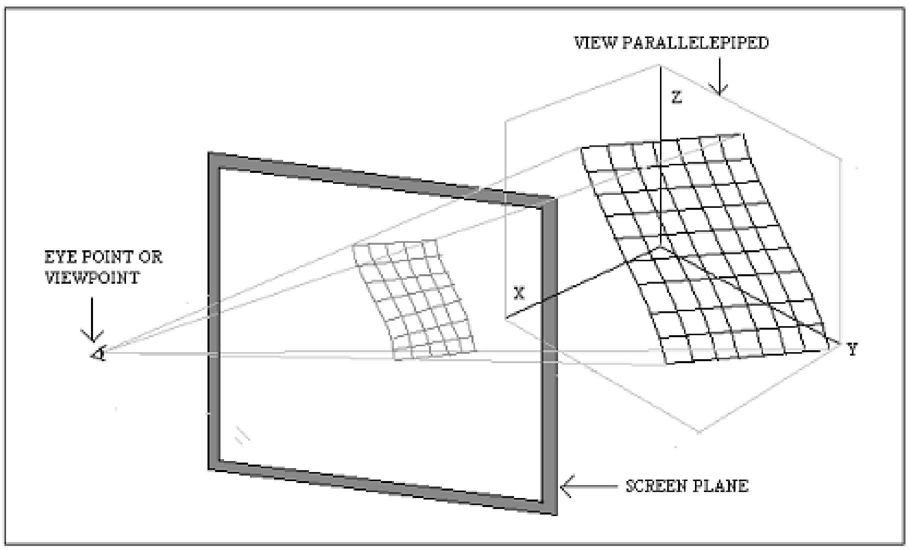
NUMX(U) und NUMY (V)
Die Funktionen NUMX und NUMY bestimmen für jede Richtung die Anzahl der Punkte bzw. der Schritte, die für die Erstellung des Bezugsrasters erforderlich sind, aus dem anschließend die Werte z = f(x,y) berechnet werden.

VPAR (W)
Dient lediglich als Referenz zu der Variable VPAR.

RESET (X)
Die Parameter auf dem Bildschirm werden auf ihre Vorgabewerte zurückgesetzt.

Drücken Sie (NXT) 3D , um zum 3D-Menü zurückzukehren.

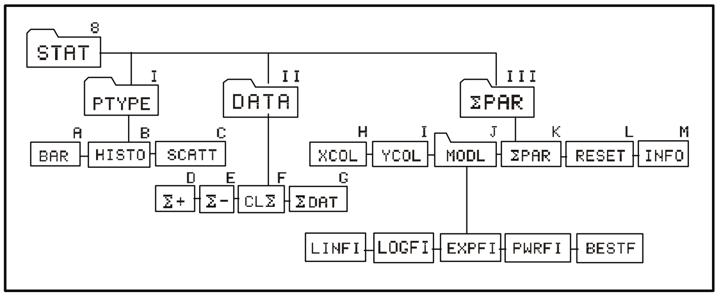
Drücken Sie PLOT, um zum PLOT-Menü zurückzukehren.



Das Menü STAT unter PLOT

Das Menü STAT ermöglicht den Zugang zu Plots, die für statistische Analysen benutzt werden. Hier stehen die folgenden Untermenüs zu Verfügung:

Die folgende Abbildung zeigt die Struktur des STAT-Menüs aus dem PLOT-Menü. Die Zahlen und Buchstaben bei jeder Funktion dienen als Referenz in der nachstehenden Beschreibung.



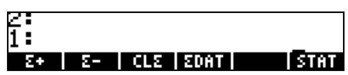
Das Menü PTYPE unter STAT (I)

Das PTYPE-Menü enthält die folgenden Funktionen:

Diese Tasten entsprechen den Plottypen *Bar (A), Histogram (B)* und *Scatter(C)*, die bereits beschrieben wurden. Wird eine dieser Funktionstasten gedrückt, während ein Programm eingegeben wird, so wird an dieser Stelle im Programm der entsprechende Funktionsaufruf eingefügt. Drücken Sie STAT, um zum STAT-Menü zurückzukehren.

Das Menü DATA unter STAT (II)

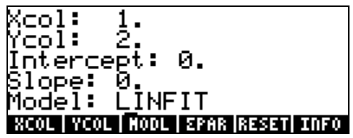
Das DATA-Menü enthält die folgenden Funktionen:



Die Funktionen in diesem Menü werden für die Manipulation der statistischen Matrix ΣDAT verwendet. Mit den Funktionen Σ+ (D) und Σ- (E) werden Datenreihen zu bzw. aus der Matrix hinzugefügt oder entfernt. CLΣ (F) löscht die ΣDAT (G) Matrix, und die Funktionstaste ΣDAT wird nur als Referenz für interaktive Anwendungen verwendet. Eine nähere Beschreibung dieser Funktionen finden Sie später, im Kapitel über statistische Anwendungen. Drücken Sie STAT, um zum STAT-Menü zurückzukehren.

Das Menü ΣPAR unter STAT (III)

Das Menü ΣPAR enthält die folgenden Funktionen:

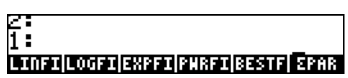


INFO (M) und ΣPAR (K)
Die Taste INFO in ΣPAR enthält die Informationen die oben abgebildet sind. Diese Informationen sind in der Variable ΣPAR zu finden. Die angezeigte Werte sind die Vorgabewerte für Spalte x, Spalte y, den Achsenabschnitt und den Richtungskoeffizienten des Daten-Angleichungsmodells, sowie der Modelltyp, der an die Daten in ΣDAT angeglichen wird.

XCOL (H)
Mit dem Befehl XCOL wird angegeben, welche Spalten aus ΣDAT, falls mehrere vorhanden, die Spalte x oder die Spalte der unabgängigen Variable sein werden.

YCOL (I)
Mit dem Befehl YCOL wird angegeben, welche Spalten aus ΣDAT, falls mehrere vorhanden, die Spalte y oder die Spalte der abgängigen Variable sein werden.

MODL (J)
Der Befehl MODL bestimmt das auszuwählende Modell für die Datenangleichung in SDAT, wenn eine Datenangleichung implementiert wird. Drücken Sie MODL, um die verfügbaren Optionen anzusehen. Das folgende Menü erscheint:



Diese Funktionen entsprechen der linearen, logarithmischen, Exponential-, Potenz- oder der besten Angleichung. Eine nähere Beschreibung der Datenangleichung finden Sie in einem Kapitel weiter unten. Drücken Sie ΣPAR, um zum ΣPAR-Menü zurückzukehren.

ΣPAR (K)
ΣPAR ist lediglich eine Referenz zu der Variable SPAR für interaktive Aktionen.

RESET (L)
Diese Funktion setzt die Inhalte von ΣPAR auf ihre Vorgabewerte zurück.

Drücken Sie NXT STAT, um zum STAT-Menü zurückzukehren. Drücken Sie [PLOT], um zum Hauptmenü PLOT zurückzukehren.

Das Menü FLAG unter PLOT

Das FLAG-Menü ist ein interaktives Menü, also steht Ihnen die Auswahl zwischen den folgenden Optionen zur Verfügung:

- AXES : fWird diese Option ausgewählt, so werden die Achsen, falls sichtbar, innerhalb der Plot-Oberfläche bzw. des Plot-Volumens angezeigt.
- CNCT : Wird diese Option ausgewählt, wird der Plot so erzeugt, dass einzelne Punkte miteinander verbunden sind.
- SIMU : Wenn diese Option ausgewählt wurde und es soll mehr als eine Grafik im gleichen Achsensystem geplottet werden, dann werden alle Grafiken gleichzeitig erzeugt.

Drücken Sie PLOT, um zum PLOT-Menü zurückzukehren.

Plots durch Programmen erzeugen

Bevor ein Plot in einem Programm erzeugt werden kann, müssen die Variablen PPAR, ΣPAR, und /oder VPAR eingestellt werden, je nachdem, ob es sich um eine zweidimensionale Grafik handelt, die mittels einer Funktion, mittels Daten aus ΣDAT oder mittels einer dreidimensionalen Funktion definiert ist. Diese Variablen können Sie mit Hilfe der bereits beschriebenen Befehle einrichten.

Als nächstes wird das allgemeine Format von Variablen beschrieben, die für die Erzeugung von verschiedenen Plottypen erforderlich sind.

Zweidimensionale Grafiken

Die zweidimensionalen Grafiken, die von den Funktionen *Function, Conic, Parametric, Polar, Truth* und *Differential Equation* erzeugt werden, verwenden die PPAR-Variable im folgenden Format:

{ (x_{min}, y_{min}) (x_{max}, y_{max}) `indep res axes ptype depend` }

Die zweidimensionalen Grafiken, die aus Daten aus der statistischen Matrix erzeugt werden, nämlich *Bar, Histogram* und *Scatter*, verwenden die ΣPAR-Variable im folgenden Format:

{ `x-column y-column slope intercept model` }

Gleichzeitig wird auch PPAR im oben angegebenen Format verwendet.

Die Bedeutung der verschiedenen Parameter aus PPAR und ΣPAR wurde im vorigen Abschnitt erläutert.

Dreidimensionale Grafiken

Die dreidimensionalen Grafiken, nämlich die Optionen *Slopefield, Wireframe, Y-Slice, Ps-Contour, Gridmap* und *Pr-Surface*, verwenden die VPAR-Variable im folgenden Format:

$\{x_{left}, x_{right}, y_{near}, y_{far}, z_{low}, z_{high}, x_{min}, x_{max}, y_{min}, y_{max}, x_{eye}, y_{eye}, z_{eye}, x_{step}, y_{step}\}$

Die Wertpaare von x, y und z haben die folgende Bedeutung:

- Dimensionen des Betrachtungsparallelepipeds ($x_{left}, x_{right}, y_{near}, y_{far}, z_{low}, z_{high}$)

- Bereich der unabhängigen Variablen x und y (x_{min}, x_{max}, y_{min}, y_{max})
- Position des Betrachtungspunktes (x_{eye}, y_{eye}, z_{eye})
- Anzahl der Schritte in den Richtungen x und y (x_{step}, y_{step})

Dreidimensionale Grafiken benötigen auch die PPAR-Variable mit den oben angegebenen Parametern.

Die Variable EQ

Außer für Plots, die auf ΣDAT basieren, müssen für alle anderen Plots die zu plottenden Funktionen definiert werden, indem die Ausdrücke oder Referenzen zu diesen Funktionen in der Variable EQ gespeichert werden.

Kurz gesagt, muss für die Erzeugung eines Plots in einem Programm gegebenenfalls die EQ-Variable geladen werden. Danach werden PPAR, PPAR und SPAR oder PPAR und VPAR geladen. Anschließend wird durch die Angabe des entsprechenden Plottyps: FUNCTION, CONIC, POLAR, PARAMETRIC, TRUTH, DIFFEQ, BAR, HISTOGRAM, SCATTER, SLOPE, WIREFRAME, YSLICE, PCONTOUR, GRIDMAP oder PARSURFACE der Plot erzeugt.

Beispiele von interaktiven Plots mit dem PLOT-Menü

Um besser zu verstehen, wie das Programm mit den PLOT-Befehlen und -Variablen umgeht, versuchen Sie die folgenden interaktiven Plots mit dem PLOT-Menü zu erzeugen.

Beispiel 1 – <u>Funktionsplot:</u>

[←] USER [F3]	PLOT-Menü (*) aufrufen
PTYPE FUNCT	FUNCTION als Plottyp wählen
'√r' [ENTER] [←] EQ	Speichern Sie die Funktion '√r' in EQ
PPAR	Plot-Parameter anzeigen
[ALPHA] [←] [R] [ENTER] INDEP	Definieren Sie 'r' als unabhängige Variable
[ALPHA] [←] [S] [ENTER] DEPND	Definieren Sie 's' als

	abhängige Variable
1 +/- SPC 10 XRNG	Definieren Sie (-1, 10) als x-Bereich
1 +/- SPC 5 YRNG NXT	Definieren Sie (-1, 5) als y-Bereich
{ (0,00 {.4 .2} "Rs" "Sr")	Achsen-Definitionsliste
AXES	Definieren Sie Achsenmitte, Ticks, Beschriftungen
NXT PLOT	Kehren Sie zum PLOT-Menü zurück
ERASE DRAX NXT LABEL	Bild löschen, Achsen, Beschriftungen zeichnen
NXT DRAW	Funktion zeichnen und Bild anzeigen
EDIT NXT MENU	Menüeinträge werden entfernt
NXT NXT PICT CANCL	Kehrt zur Normalanzeige zurück

(*) Das Plot-Menü kann durch die benutzerdefinierte Taste F3 aufgerufen werden, wie in diesem Kapitel bereits beschrieben wurde.

Beispiel 2 – <u>Parametrischer Plot:</u>

← USER F3	PLOT-Menü aufrufen
PTYPE PARAM	PARAMETRIC als Plottyp wählen
{ 'SIN(t)+i*SIN(2*t)' } ENTER	Definieren Sie die komplexe Funktion X+iY
← EQ	Speichern Sie die komplexe Funktion in EQ
PPAR	Plot-Parameter anzeigen
{t 0 6.29} ENTER INDEP	Definieren Sie 't' als unabhängige Variable
ALPHA Y ENTER DEPND	Definieren Sie 'Y' als abhängige Variable
2.2 +/- SPC 2.2 XRNG	Definieren Sie (-2.2, 2.2) als x-Bereich
1.1 +/- SPC 1.1 YRNG NXT	Definieren Sie (-1.1, 1.1) als y-Bereich
{ (0,0) {.4 .2} "X(t)" "Y(t)"} ENTER	Achsen-Definitionsliste
AXES	Definieren Sie Achsenmitte, Ticks, Beschriftungen
NXT PLOT	Kehren Sie zum PLOT-Menü

	zurück
ERASE DRAX NXT LABEL	Bild löschen, Achsen, Beschriftungen zeichnen
NXT DRAW	Funktion zeichnen und Bild anzeigen
EDIT NXT MENU NXT NXT PICT CANCL	Plot beenden

Beispiel 3 –<u>Polarplot:</u>

← USER F3	PLOT-Menü aufrufen
PTYPE POLAR	POLAR als Plottyp wählen
'1+SIN(θ)' ENTER ← EQ	Speichern Sie die komplexe Funktion r = f(θ) in EQ
PPAR	Plot-Parameter anzeigen
{ θ 0 6.29} ENTER INDEP	Definieren Sie 'θ' als unabhängige Variable
ALPHA Y ENTER DEPND	Definieren Sie 'Y' als abhängige Variable
3 +/- SPC 3 XRNG	Definieren Sie (-3,3) als x-Bereich
0.5 +/- SPC 2.5 YRNG NXT	Definieren Sie (-0.5, 2.5) als y-Bereich
{ (0,0) {.5 .5} "x" "y"} ENTER	Achsen-Definitionsliste
AXES	Definieren Sie Achsenmitte, Ticks, Beschriftungen
NXT PLOT	Kehren Sie zum PLOT-Menü zurück
ERASE DRAX NXT LABEL	Bild löschen, Achsen, Beschriftungen zeichnen
NXT DRAW	Funktion zeichnen und Bild anzeigen
EDIT NXT MENU	Menüeinträge entfernen
NXT NXT PICT CANCL	Kehrt zur Normalanzeige zurück

Diese Beispiele zeigen ein Muster für die interaktive Erzeugung von zweidimensionalen Grafiken durch das PLOT-Menü.

1 – PTYPE auswählen
2 – Die zu plottende Funktion in der Variablen EQ speichern (dabei auf das richtige Format achten, z.B. 'X(t)+iY(t)' bei PARAMETRIC)

3 – Namen (und gegebenenfalls den Bereich) der unabhängigen und abhängigen Variablen eingeben
4 – Die Angaben für die Achse als Liste eingeben { center atick x-label y-label }
5 – Mit ERASE, DRAX, LABEL, DRAW eine vollständig beschriftete Grafik mit Achsen erzeugen

Auf gleiche Weise können Plots mit einem Programm erzeugt werden, aber in diesem Fall muss nach dem Aufruf der Funktion DRAW auch der Befehl PICTURE eingefügt werden, um den Grafikbildschirm wieder in den Stack zu laden.

Beispiele von programm-generierten Plots

In diesem Abschnitt werden wir Ihnen zeigen, wie die Plots aus den letzten drei Beispielen mit einem Programm erzeugt werden. Bevor Sie mit der Eingabe des Programms beginnen, aktivieren Sie das PLOT-Menü, um die Eingabe von Grafikbefehlen (wie z.B. (← USER F3) zu erleichtern.

Beispiel 1 –<u>Funktionsplot:</u> Geben Sie das folgende Programm ein:

`«`	Programm starten
`{PPAR EQ} PURGE`	Die aktuellen Werte von PPAR und EQ löschen
`'√r' STEQ`	'√r' in EQ speichern
`'r' INDEP`	Unabhängige Variable auf 'r' setzen
`'s' DEPND`	Abhängige Variable auf 's' setzen
`FUNCTION`	FUNCTION als Plottyp wählen
`{ (0.,0.) {.4 .2}`	
`"Rs" "Sr" } AXES`	Achseninformationen setzen
`-1. 5. XRNG`	x-Bereich setzen
`-1. 5. YRNG`	y-Bereich setzen
`ERASE DRAW DRAX LABEL`	Bild löschen und Plot, Achsen und Beschriftungen zeichnen
`PICTURE »`	Grafikbildschirm wieder in den Stack laden

Speichern Sie das Programm in der Variable PLOT1. Um das Programm auszuführen, drücken Sie VAR, falls erforderlich, und anschließend drücken Sie PLOT1.

Beispiel 2 – <u>Parametrischer Plot:</u> Geben Sie das folgende Programm ein:

`«`	Programm starten
`RAD {PPAR EQ} PURGE`	Auf Radianten wechseln, Variablenwerte löschen
`'SIN(t)+i*SIN(2*t)' STEQ`	'X(t)+iY(t)' in EQ speichern
`{ t 0. 6.29} INDEP`	Unabhängige Variable auf 'r' setzen, mit Bereich
`'Y' DEPND`	Abhängige Variable auf 'Y' setzen
`PARAMETRIC`	PARAMETRIC als Plottyp wählen
`{ (0.,0.) {.5 .5} "X(t)"`	
`"Y(t)" } AXES`	Achseninformationen setzen
`-2.2 2.2 XRNG`	x-Bereich setzen
`-1.1 1.1 YRNG`	y-Bereich setzen
`ERASE DRAW DRAX LABEL`	Bild löschen und Plot, Achsen und Beschriftungen zeichnen
`PICTURE`	Grafikbildschirm wieder in den Stack laden
`»`	Programm beenden

Speichern Sie das Programm in der Variable PLOT2. Um das Programm auszuführen, drücken Sie VAR, falls erforderlich, und drücken Sie anschließend PLOT2.

Beispiel 3 –<u>Polarplot:</u> Geben Sie das folgende Programm ein:

`«`	Programm starten
`RAD {PPAR EQ} PURGE`	Auf Radianten wechseln, Variablenwerte löschen
`'1+SIN(θ)' STEQ`	'f(θ)' in EQ speichern
`{ θ 0. 6.29} INDEP`	Unabhängige Variable auf 'θ' setzen, mit Bereich
`'Y' DEPND`	Abhängige Variable auf 'Y' setzen

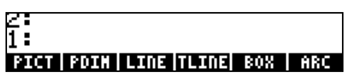
`POLAR`	POLAR als Plottyp wählen
`{ (0.,0.) {.5 .5}`	
`"x" "y"} AXES`	Achseninformationen setzen
`-3.  3. XRNG`	x-Bereich setzen
`-.5  2.5 YRNG`	y-Bereich setzen
`ERASE DRAW DRAX LABEL`	Bild löschen und Plot, Achsen und Beschriftungen zeichnen
`PICTURE`	Grafikbildschirm wieder in den Stack laden
`»`	Programm beenden

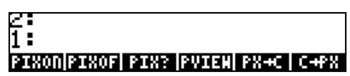
Speichern Sie das Programm in der Variable PLOT3. Um das Programm auszuführen, drücken Sie VAR, falls erforderlich, und drücken Sie anschließend PLOT3.

Diese Beispiele zeigen die Verwendung von PLOT-Befehlen in Programmen. Hierbei handelt es sich nur um einfache Übungen für die Programmierung von Plots. Sie sollten versuchen, eigene Übungen in der Programmierung von Plots zu machen.

Zeichenbefehle für die Programmierung

Sie können Abbildungen in einem Grafikfenster direkt aus einem Programm einfügen, mit Hilfe von Befehlen wie z.B. die, aus dem PICT-Menü, die mit ← PRG NXT PICT aufgerufen werden. In diesem Menü stehen die folgenden Funktionen zur Verfügung: Drücken Sie NXT , um zum nächsten Menü zu gelangen.





Logischerweise führen die Befehle LINE, TLINE und BOX die gleichen Operationen aus, wie die gleichen interaktiven Befehle, vorausgesetzt es werden die selben Angaben gemacht. Diese, sowie die anderen Funktionen im PICT-Menü beziehen sich auf das Grafikfenster, in dem die Bereiche von x und y in der Variable PPAR bestimmt werden, wie dies bereits für verschiedene Typen von Grafiken demonstriert wurde. Nachstehend werden die Funktionen des PICT-Befehls beschrieben.

PICT

Diese Funktionstaste entspricht der Variable PICT, in der die aktuellen Inhalte des Grafikfensters gespeichert werden. Der Name der Variable kann jedoch nicht zwischen Anführungszeichen gesetzt werden und in der Variable können nur Grafikobjekte gespeichert werden. Aus diesem Grund unterscheidet sich PICT von allen anderen Rechnervariablen.

PDIM

Die Funktion PDIM erfordert die Eingabe von entweder zwei geordneten Paaren (x_{min},y_{min}) (x_{max} y_{max}) oder von zwei binären Ganzzahlen #w und #h. PDIM bewirkt, dass die aktuelle Inhalte von PICT durch einen leeren Bildschirm ersetzt werden. Wenn das Argument (x_{min},y_{min}) (x_{max} y_{max}) ist, werden diese Werte zum Bereich der benutzerdefinierten Koordinaten in PPAR. Wenn das Argument #w und #h ist, bleiben die Bereiche der benutzerdefinierten Koordinaten in PPAR unverändert, aber die Größe der Grafik ändert sich auf #h × #w Pixel.

PICT und der Grafikbildschirm

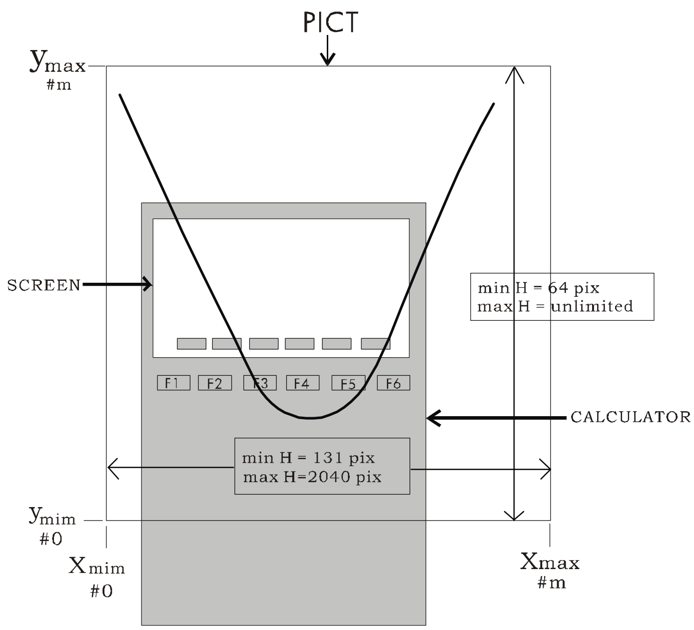
PICT, der Speicherbereich der aktuellen Grafik, kann als eine zweidimensionale Grafik betrachtet werden, die eine Mindestgröße von 131 Pixel breit mal 64 Pixel hoch hat. Die maximale Breite von PICT beträgt 2048 Pixel, ohne Beschränkung der maximalen Höhe. Die Pixel sind die einzelnen Punkte des Bildschirms, die ein- oder ausgeschaltet (dunkel bzw. hell) werden, um Text oder Grafiken darzustellen. Der Bildschirm des Rechners besteht aus 131 mal 64 Pixel, was der Mindestgröße von PICT entspricht. Falls PICT größer ist als der Bildschirm, kann die PICT-Grafik als eine zweidimensionale Domäne betrachtet werden, die auf dem Bildschirm bewegt werden kann, wie in der nachstehenden Abbildung zu sehen ist.

LINE

Dieser Befehl erfordert als Eingabe zwei geordnete Paare (x_1,y_1) (x_2, y_2) oder zwei Paare von Pixelkoordinaten {#n_1 #m_1} {#n_2 #m_2}. Der Befehl zeichnet eine Linie zwischen den angegebenen Koordinaten.

TLINE

Dieser Befehl (Toggle LINE – umgekehrte Linie) erfordert als Eingabe zwei geordnete Paare (x_1,y_1) (x_2, y_2) oder zwei Paare von Pixelkoordinaten {#n_1 #m_1} {#n_2 #m_2}. Der Befehl zeichnet eine Linie zwischen den angegebenen Koordinaten und schaltet anschließend die eingeschalteten Pixel der Linie aus und umgekehrt.



BOX

Dieser Befehl erfordert als Eingabe zwei geordnete Paare (x_1,y_1) (x_2, y_2) oder zwei Paare von Pixelkoordinaten {#n_1 #m_1} {#n_2 #m_2}. Der Befehl zeichnet ein Kästchen, bei dem die Diagonalen von den zwei eingegebenen Koordinatenpaaren bestimmt werden.

ARC

Mit diesem Befehl wird ein Bogen gezeichnet. ARC erfordert die folgenden Objekte als Eingabe:

- Koordinaten der Bogenmitte als (x,y) in benutzerdefinierten Koordinaten oder {#n, #m} in Pixel.
- Radius des Bogens als r (benutzerdefinierte Koordinaten) oder #k (Pixel).
- Anfangswinkel θ_1 und Endwinkel θ_2.

PIX?, PIXON und PIXOFF

Diese Funktionen erfordern als Eingabe die Koordinaten des Punktes in Benutzereinheiten (x,y) oder in Pixel {#n, #m}.

- PIX? überprüft, ob an der Position (x,y) bzw. {#n, #m} der Pixel an ist.
- PIXOFF schaltet den Pixel an der Position (x,y) bzw. {#n, #m} aus.
- PIXON schaltet den Pixel an der Position (x,y) bzw. {#n, #m} ein.

PVIEW

Dieser Befehl erfordert als Eingabe die Koordinaten eines Punktes in Benutzereinheiten (x,y) oder in Pixel {#n, #m}, und setzt die Inhalte aus PICT mit der linken oberen Ecke an den angegebenen Punkt. Wenn eine leere Liste als Argument angegeben wird, wird das Bild in die Bildschirmmitte gesetzt. PVIEW aktiviert den Grafikcursor bzw. das Bildmenü nicht. Diese können mit PICTURE aktiviert werden.

PX→C

Die Funktion PX→C konvertiert Pixelkoordinaten {#n #m} in Koordinaten in Benutzereinheiten (x,y).

C→PX

Die Funktion C→PX konvertiert Koordinaten in Benutzereinheiten (x,y) in Pixelkoordinaten {#n #m}.

Programmierbeispiele mit Zeichenfunktionen

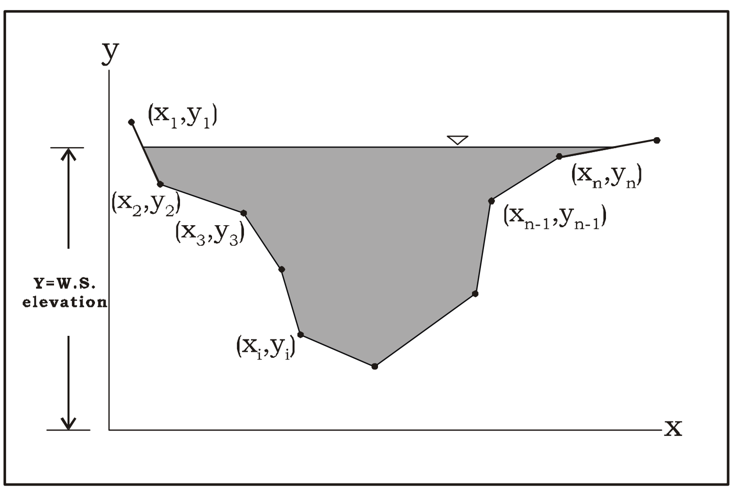
In diesem Abschnitt werden die oben beschriebenen Befehle für die Erzeugung von Grafiken mit Hilfe von Programmen verwendet. Die Programmcodes sind auf der mitgelieferten Diskette bzw. CD-ROM zu finden.

Beispiel 1 – Ein Programm, das Zeichenbefehle verwendet
Das folgende Programm erzeugt eine Zeichnung auf dem Grafikbildschirm. (Dieses Programm wurde nur dafür geschrieben, die Befehle für die Erzeugung von Zeichnungen auf dem Bildschirm zu erläutern).

«	Programm starten
DEG	Wählen Sie Grad für Winkelmaße
0. 100. XRNG	x-Bereich setzen
0. 50. YRNG	y-Bereich setzen
ERASE	Bild löschen
(5., 2.5) (95., 47.5) BOX	Kästchen zwischen (5,5) und (95,95) zeichnen
(50., 50.) 10. 0. 360. ARC	Kreis mit der Mitte in (50,50) und r =10 zeichnen
(50., 50.) 12. –180. 180. ARC	Kreis mit der Mitte in (50,50) und r =12 zeichnen
1 8 FOR j	8 Linien im Inneren des Kreises zeichnen
(50., 50.) DUP	Die Linienmitte befindet sich im Punkt (50,50)
'12*COS(45*(j-1))' →NUM	x berechnen, das andere Ende bei 50 + x
'12*SIN(45*(j-1))' →NUM	y berechnen, das andere Ende bei 50 + y
R → C	x y in (x,y) konvertieren, komplexe Zahlen
+	(50,50) mit (x,y) addieren
LINE	Linie zeichnen
NEXT	Ende der FOR-Schleife
{ } PVIEW	Bild anzeigen
»	

Beispiel 2 – Ein Programm zum Plotten eines Flussquerschnittes
Diese Anwendung kann sich bei der Feststellung der Fläche und der befeuchteten Randbereiche eines Flussquerschnittes als sehr nützlich herausstellen. Normalerweise wird ein Flussquerschnitt vermessen und daraus resultiert eine Reihe von Punkten, und zwar die Koordinaten x und y bezogen auf ein beliebiges Koordinatensystem. Diese Punkte werden geplottet und danach wird eine Entwurfszeichnung des Querschnitts in einer bestimmte Höhe der Wasseroberfläche erzeugt. Die folgende Abbildung zeigt die hier beschriebenen Tatsachen.

Dieses Programm, das auf der mitgelieferten Diskette bzw. CD-ROM zu finden ist, verwendet vier Unterprogramme, und zwar FRAME, DXBED, GTIFS und INTRP. Das Hauptprogramm, genannt XSECT, erfordert als Eingabe eine Matrix mit den Werten von x und y, sowie die Höhe der Wasseroberfläche Y (siehe Abbildung), in dieser Reihenfolge. Das Programm erzeugt eine Grafik des Querschnittes, wobei die Eingabedaten auf der Grafik als Punkte dargestellt werden. Außerdem wird die freie Oberfläche des Querschnittes angezeigt.



Wir empfehlen, dass Sie ein neues Unterverzeichnis für die Speicherung der Programme erstellen. Sie können das Unterverzeichnis FLUSS benennen, da es sich hier um unregelmäßigen Querschnitten mit offenen Kanälen handelt, die für Flüsse typisch sind.

Um das Programm XSECT auszuführen, verwenden Sie die folgenden Datensätze. Diese werden als Matrizen mit zwei Spalten eingegeben, wobei die erste Spalte für x und die zweite für y vorgesehen ist. Die Matrizen werden in Variablen mit Namen wie z.B. XYD1 (X-Y Datensatz 1) und XYD2 (X-Y Datensatz 2) gespeichert. Um das Programm auszuführen, legen Sie einen Datensatz in den Stack (z.B. VAR XYD1), geben Sie danach die Höhe der Wasseroberfläche (z.B. 4.0) ein, und drücken Sie anschließend XSECT. Der Rechner zeigt eine Entwurfszeichnung des Querschnittes mit der entsprechenden Wasseroberfläche an. Um die Grafikanzeige zu beenden, drücken Sie ON.

Versuchen Sie folgende Beispiele:

XYD1 2 XSECT
XYD1 3 XSECT
XYD1 4 XSECT
XYD1 6 XSECT

Bei der Ausführung des Programms XSECT müssen Sie ein wenig Geduld haben. Aufgrund der hohen Anzahl von Grafikfunktionen, ohne die numerischen Iterationen mitzuzählen, kann es eine Weile dauern, bis die Grafik erzeugt wird (etwa 1 Minute).

Datensatz 1

x	y
0.4	6.3
1.0	4.9
2.0	4.3
3.4	3.0
4.0	1.2
5.8	2.0
7.2	3.8
7.8	5.3

Datensatz 2

x	y
0.7	4.8
1.0	3.0
1.5	2.0
2.2	0.9
3.5	0.4
4.5	1.0
5.0	2.0
6.0	2.5

9.0	7.2		7.1	2.0
			8.0	0.7
			9.0	0.0
			10.0	1.5
			10.5	3.4
			11.0	5.0

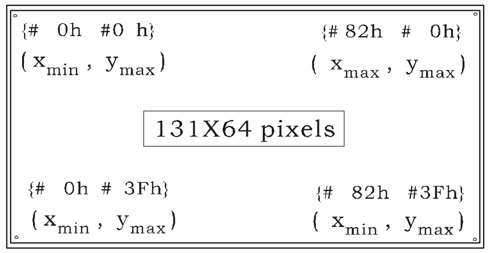
Anmerkung: Das Programm FRAME, wie es ursprünglich geschrieben wurde (siehe Diskette oder CD-ROM) erhält den richtigen Maßstab der Grafik nicht. Wenn Sie den richtigen Maßstab erhalten möchten, ersetzen Sie FRAME durch das folgende Programm:

```
« STOΣ MINΣ MAXΣ 2 COL→ DUP →COL DROP – AXL ABS AXL 20
/ DUP NEG SWAP 2 COL→ + →ROW DROP SWAP → yR xR « 131
DUP R→B SWAP yR OBJ→ DROP – xR OBJ→ DROP - / * FLOOR
R→B PDIM yR OBJ→ DROP YRNG xR OBJ→ DROP XRNG ERASE » »
```

Dieses Programm beschränkt die Breite der PICT-Variable auf 131 Pixel – die minimale Pixelgröße der horizontalen Achse – und passt die Anzahl der Pixel der vertikalen Achse so an, dass der 1:1 Maßstab zwischen der vertikalen und der horizontalen Achse erhalten bleibt.

Pixelkoordinaten

Die folgende Abbildung zeigt die Grafikkoordinaten für einen typischen (minimalen) Bildschirm von 131×64 Pixeln. Pixelkoordinaten werden von der linken oberen Ecke des Bildschirms {# 0h # 0h} gemessen. Dies entspricht den benutzerdefinierten Koordinaten (x_{min}, y_{max}). Die maximalen Koordinaten in Pixel entsprechen der rechten unteren Ecke des Bildschirms {# 82h #3Fh}; dieser Punkt entspricht den benutzerdefinierten Koordinaten (x_{max}, y_{min}). Die Koordinaten der zwei Ecken werden in der nachfolgenden Abbildung sowohl in Pixel als auch in benutzerdefinierten Koordinaten dargestellt.

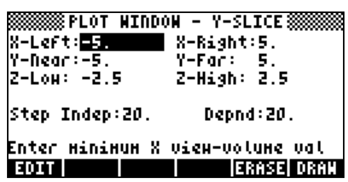


Grafiken animieren

Hier werden Sie erfahren, wie Sie mit dem Plottyp Y-Slice animierte Grafiken erzeugen können. Nehmen wir an, dass Sie die Wanderwelle f(X,Y) = 2.5 sin(X-Y) animieren wollen. Wir können X in der Animation des Plots der Funktion f(X,Y) gegen Y als Zeit mit verschiedenen Werten für X betrachten. Gehen Sie wie folgt vor, um die Grafik zu erstellen:

- ↰ 2D/3D gleichzeitig drücken. Wählen Sie Y-Slice für TYPE. '2.5*SIN(X-Y)' für EQ. 'X' für INDEP. Drücken Sie NXT OK.

- ↰ WIN gleichzeitig drücken (im RPN-Modus). Verwenden Sie die folgenden Werte:



- Drücken Sie ERASE DRAW. Lassen Sie dem Rechner ein wenig Zeit, damit er alle erforderlichen Grafiken erzeugen kann. Wenn die Operation abgeschlossen ist, wird auf dem Bildschirm eine sinusförmige Wanderwelle angezeigt.

Animation von Grafiksammlungen

Mit der Funktion ANIMATE kann eine Reihe von Grafiken, die in den Stack abgelegt wurden, animiert werden. Sie können eine Grafik auf dem Grafikbildschirm erzeugen, wenn Sie die Befehle aus den Menüs PLOT und PICT verwenden. Um die erzeugte Grafik in den Stack abzulegen, verwenden Sie PICT RCL. Wenn Sie *n* Grafiken auf den Ebenen *n* bis *1* des Stacks haben, können Sie mit dem Befehl *n* ANIMATE aus den Grafiken im Stack eine Animation erstellen.

<u>Beispiel 1</u> – Animation einer Welle auf der Wasseroberfläche
Als Beispiel geben Sie das folgende Programm ein, das 11 Grafiken erzeugt, und zwar so, dass in der Mitte des Grafikbildschirms ein Kreis abgebildet wird und der Radius dieses Kreises wird in jeder nachfolgenden Grafik um einen konstanten Wert erhöht.

«	Programm starten
RAD	Winkeleinheit auf Radianten setzen
131 R→B 64 R→B PDIM	PICT auf 131x64 Pixel setzen
0 100 XRNG 0 100 YRNG	x- und y-Bereiche auf 0-100 setzen
1 11 FOR j	Schleife bei j = 1 starten 11
ERASE	Aktuelles PICT löschen
(50., 50.) '5*(j-1)' →NUM	Die Mitte der Kreise (50,50)
0 '2*π' →NUM ARC	Kreis mit einem Radius von r = 5(j-1) zeichnen
PICT RCL	Aktuelles PICT in den Stack ablegen
NEXT	Ende der FOR-NEXT-Schleife
11 ANIMATE	Animieren
»	Programm beenden

Speichern Sie dieses Programm in der Variable PANIM (Plot ANIMation). Um das Programm auszuführen, drücken Sie VAR (falls erforderlich) und PANIM. Der Rechner braucht mehr als eine Minute, um die Grafiken zu erzeugen und die Animation zu starten. Aus diesem Grund müssen Sie sich ein wenig gedulden. Auf dem Bildschirm erscheint das Sanduhr-Symbol für eine scheinbar längere Zeit, und danach erscheint auf dem Bildschirm eine Animation, die an Wellen erinnert, die sich beim Eintauchen eines kleinen Steines auf der Wasseroberfläche bilden. Drücken Sie ON, um die Animation zu beenden.

Die 11 Grafiken, die vom Programm erzeugt wurden, stehen im Stack weiterhin zur Verfügung. Wenn Sie die Animation erneut ausführen möchten, verwenden Sie einfach: 11 ANIMATE. (Die Funktion ANIMATE können Sie mit ← PRG NXT GROB NXT ANIMA aufrufen). Die Animation wird erneut ausgeführt. Drücken Sie ON, um die Animation wieder zu beenden. Beachten Sie, dass die Zahl 11 weiterhin auf Ebene 1 des Stacks bleibt. Drücken Sie ◄, um sie aus dem Stack zu entfernen.

Nehmen wir an, dass Sie die Zahlen, die diese Animation bilden, in einer Variable speichern wollen. Sie können eine Liste, sagen wir WLIST, mit diesen Zahlen zusammenstellen, wenn Sie wie folgt vorgehen:

1 1 ← PRG TYPE →LIST ' ALPHA ALPHA W L I S T ALPHA STO►

Drücken Sie VAR, um die Liste mit den Variablen wiederherzustellen. Die Variable WLIST sollte jetzt in der Auflistung der Funktionstasten aufgenommen werden. Um die Variablenliste erneut zu animieren, verwenden Sie das folgende Programm:

«	Programm starten
WLIST	WLIST-Liste in den Stack ablegen
OBJ→	Liste zerlegen, Stack-Ebene 1 = 11
ANIMATE	Animation starten
»	Programm beenden

Speichern Sie dieses Programm in der Variable RANIM (Re-ANIMate).
Drücken Sie RANIM, um das Programm auszuführen.

Das folgende Programm animiert die Grafiken aus WLIST vor und zurück:

«	Programm starten
WLIST DUP	WLIST-Liste in den Stack ablegen, eine Kopie machen
REVLIST +	Reihenfolge umkehren, die 2 Listen verknüpfen
OBJ→	Liste in seine Elemente zerlegen, Ebene 1 = 22
ANIMATE	Animation starten
»	Programm beenden

Speichern Sie dieses Programm in der Variable RANI2 (Re-ANImate Version 2). Drücken Sie RANI2, um das Programm auszuführen. Die Animation simuliert jetzt eine Welle auf der Wasseroberfläche, die von den Wänden eines kreisförmigen Behälters zur Mitte reflektiert wird. Drücken Sie ON, um die Animation zu beenden.

<u>Beispiel 2</u> - Animation von Potenzfunktionen
Nehmen wir an, Sie wollen den Plot der Funktionen $f(x) = x^n$, n = 0, 1, 2, 3, 4 im selben Achsensystem animieren. Sie können das folgende Programm verwenden:

«	Programm starten
RAD	Winkeleinheit auf Radianten setzen
131 R→B 64 R→B PDIM	PICT auf 131×64 Pixel setzen
0 2 XRNG 0 20 YRNG	x- und y-Bereiche setzen
0 4 FOR j	Schleife wird bei j = 0,1,…,4 gestartet
'X^j' STEQ	'X^j' in der Variable EQ ablegen
ERASE	Aktuelles PICT löschen
DRAX LABEL DRAW	Achsen, Beschriftungen, Funktion zeichnen
PICT RCL	Aktuelles PICT in den Stack ablegen
NEXT	Ende der FOR-NEXT-Schleife
5 ANIMATE	Animieren
»	

Speichern Sie dieses Programm in der Variable PWAN (PoWer function Animation – Animation der Potenzfunktion). Um das Programm auszuführen, drücken Sie VAR (falls erforderlich) und PWAN. Der Rechner zeichnet jede einzelne Potenzfunktion bevor die Animation, in der die fünf Funktionen eine nach der anderen geplottet werden, gestartet wird. Drücken Sie ON, um die Animation zu beenden.

Weitere Informationen zu der Funktion ANIMATE

Die Funktion ANIMATE, wie sie in den zwei vorigen Beispielen verwendet wurde, benötigt als Eingabe die zu animierende Grafiken sowie die Anzahl der Grafiken. Sie können zusätzliche Informationen für die Animation angeben, wie z.B. das Zeitintervall zwischen der Darstellung der Grafiken und die Anzahl der Wiederholungen der Darstellung. Das allgemeine Format der Funktion ANIMATE für solche Fälle ist wie folgt:

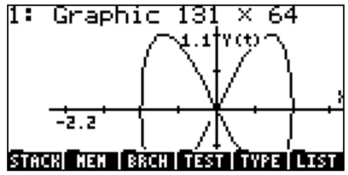
```
n-graphs   { n {#X #Y} delay rep }  ANIMATE
```

`n` ist die Anzahl der Grafiken, `{#X #Y}` steht für die Pixelkoordinaten der rechten unteren Ecke der zu plottenden Fläche (siehe Abbildung unten), `delay` sind die Sekunden zwischen den Darstellungen von aufeinander folgenden Grafiken der Animation und `rep` ist die Anzahl der Wiederholungen der Animation.

Grafikobjekte (GROBs)

Das Wort GROB kommt von GRaphics OBjects (Grafikobjekte) und es entspricht der Pixel-zu-Pixel-Beschreibung eines Bildes auf dem Bildschirm. Aus diesem Grund, wenn ein Bild in einem GROB konvertiert wird, wird aus diesem eine Reihe von binären Ziffern (m Englisch *b*inary dig*its* = *bits*), d.h. Nullen und Einsen. Nehmen wir das folgende Beispiel, um die GROBs, sowie die Konvertierung von Bildern in GROBs zu erläutern.

Wenn eine Grafik auf dem Rechner erstellt wird, wird diese Grafik als Inhalt einer besonderen Variable PICT gespeichert. Der Inhalt von PICT kann mit dieser Tastenfolge angezeigt werden: PICT RCL (← PRG NXT PICT PICT ← RCL).
Auf Ebene 1 des Stacks wird line Graphic 131 64 angezeigt (falls die standardmäßige Bildschirmgröße verwendet wird), gefolgt von einer Entwurfszeichnung des oberen Teils der Grafik. Beispiel:



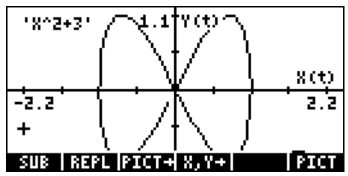
Wenn [▼] gedrückt wird, wird die Grafik aus Ebene 1 auf dem Bildschirm des Rechners angezeigt. Drücken Sie [CANCL], um zur Normalanzeige des Rechners zurückzukehren.

Die Grafik auf Ebene 1 ist immer noch nicht im GROB-Format, obwohl sie definitionsmäßig ein Grafikobjekt ist. Um die Grafik aus dem Stack in ein GROB zu konvertieren, müssen Sie die folgende Eingabe vornehmen: [3] [ENTER] [←] [PRG] [NXT] [GROB] [→GROB]. Jetzt haben wir die folgenden Informationen auf Ebene 1:

Der erste Teil der Beschreibung ist ähnelt dem, was wir ursprünglich hatten, und zwar `Graphic 131×64`, aber diesmal haben wir `Graphic 13128 × 8`. In diesem Fall wird die Grafikanzeige jedoch von einer Reihe von Nullen und Einsen ersetzt, welche die Pixel der Originalgrafik darstellen. Auf dieser Weise wurde jetzt die Originalgrafik in die entsprechende Bit-Darstellung konvertiert.

Sie können auch Gleichungen in GROBs konvertieren. Schreiben Sie mit dem EquationWriter die Gleichung 'X^2+3' auf Ebene 1 des Stacks und drücken Sie anschließend
[1] [ENTER] [←] [PRG] [NXT] [GROB] [→GROB] . Auf Ebene 1 haben Sie jetzt ein GROB mit der folgenden Beschreibung:

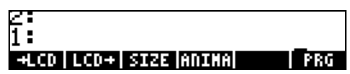
Diese zum Grafikobjekt konvertierte Gleichung kann jetzt in der Grafikanzeige angesehen werden. Bum die Grafik wiederherzustellen, drücken Sie ◄ . Bewegen Sie den Cursor auf einer leeren Stelle in der Grafik und drücken Sie EDIT NXT NXT REPL. Die Gleichung 'X^2-5' wird in die Grafik eingefügt. Beispiel:



Auf dieser Art können Sie GROBs für die Dokumentation von Grafiken verwenden, indem Sie Gleichungen oder Texte in die Grafikanzeige einfügen.

Das Menü GROB

Das Menü GROB kann mit ← PRG NXT GROB →GROB aufgerufen werden und enthält die folgenden Funktionen. Drücken Sie NXT , um sich zum nächsten Menü zu bewegen.



→GROB

Einige dieser Funktionen haben wir bereits verwendet: SUB, REPL (aus dem Grafikmenü EDIT), ANIMATE [ANIMA] und →GROB. ([*PRG*] ist nur eine Möglichkeit, zum Programmmenü zurückzukehren.) Sie haben bestimmt bemerkt, dass in den letzten zwei Beispielen bei der Konvertierung der Grafik in ein GROB die Zahl 3 verwendet wurde, wobei bei der Konvertierung der Gleichung in ein GROB die Zahl 1 verwendet wurde. Dieser Parameter der Funktion →GROB gibt die Größe des zu konvertierenden Objektes an, und zwar 0 oder 1 für kleine Objekte, 2 für mittelgroße Objekte und 3 für große Objekte. Nachstehend werden die übrigen Funktionen des GROB-Menüs beschrieben.

BLANK

Die Funktion BLANK, mit den Argumenten #n und #m, erstellt ein leeres Grafikobjekt mit der Breite und Höhe, die durch die Werte #n bzw. #m angegeben wird. Diese Funktion ist vergleichbar mit der PDIM-Funktion aus dem GRAPH-Menü.

GOR

Die Funktion GOR (Graphics OR) erfordert als Eingabe $grob_2$ (ein Ziel-GROB), ein Koordinatensystem und $grob_1$. Diese Funktion bewirkt die Überlagerung von $grob_1$ und $grob_2$ (oder PICT), beginnend bei den angegebenen Koordinaten. Die Koordinaten können in Benutzereinheiten (x,y) oder in Pixel {#n #m} angegeben werden. GOR verwendet die OR-Funktion, um den Zustand (an oder aus) jedes Pixels im überlappenden Bereich zwischen $grob_1$ und $grob_2$ zu ermitteln.

GXOR

Die Funktion GXOR (Graphics XOR) arbeitet wie GOR, aber in diesem Fall wird der finale Zustand der Pixel im überlappenden Bereich zwischen den Objekten $grob_1$ und $grob_2$ mit XOR ermittelt.

Anmerkung: Wenn in GOR oder GXOR *grob2* durch PICT ersetzt wird, erfolgt keine Ausgabe. Um die Ausgabe anzusehen, müssen Sie PICT mit PICT RCL oder PICTURE wieder in den Stack laden.

→LCD

Nimmt das angegebene GROB und zeigt es auf dem Bildschirm beginnend in der linken obere Ecke.

LCD→

Kopiert den Inhalt des Stacks und der Menüanzeige in ein GROB von 131 x 64 Pixeln.

SIZE

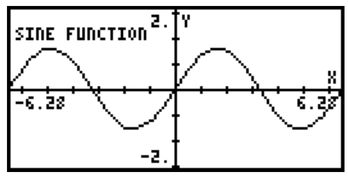
Die Funktion SIZE, wenn sie auf ein GROB angewendet wird, zeigt die Größe des GROBs in der Form von zwei Zahlen. Die erste Zahl (auf Ebene 2 im Stack) ist die Breite des Grafikobjektes, die zweite Zahl (auf Ebene 1) ist die Höhe.

Beispiel eines Programms mit GROB

Das folgende Programm erzeugt die Grafik einer Sinusfunktion, einschließlich eines Rahmens, der mit der BOX-Funktion gezeichnet wird, und eines GROBs, der als Beschriftung dient. Nachfolgend ist der Programmcode aufgeführt:

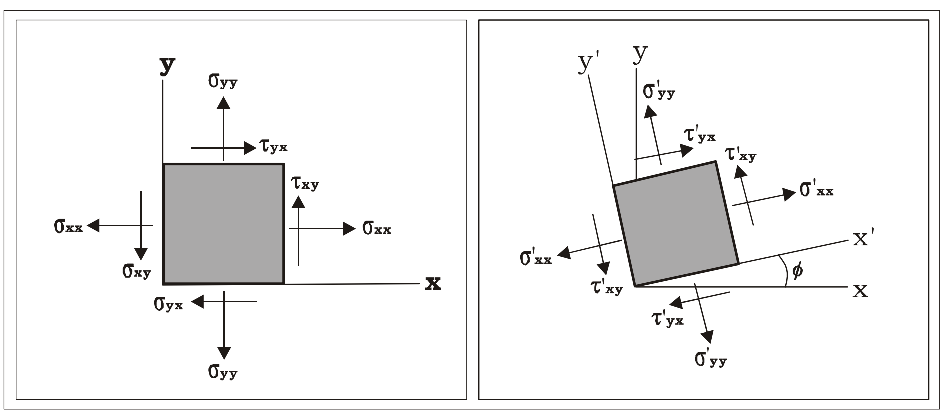
«	Programm starten
RAD	Winkeleinheit auf Radianten setzen
131 R→B 64 R→B PDIM	PICT auf 131×64 Pixel setzen
-6.28 6.28 XRNG –2. 2. YRNG	x- und y-Bereiche setzen
FUNCTION	FUNCTION als Grafiktyp auswählen
'SIN(X)' STEQ	Sinusfunktion in EQ speichern
ERASE DRAX LABEL DRAW	Löschen und dann Achsen, Beschriftungen, Grafik zeichnen
(-6.28,-2.) (6.28,2.) BOX	Rahmen um die Grafik zeichnen
PICT RCL	Inhalt von PICT in den Stack ablegen
"SINE FUNCTION"	Zeichenkette für die Grafikbeschriftung in den Stack legen
1 →GROB	Zeichenkette in kleinen GROB konvertieren
(-6., 1.5) SWAP	Koordinaten des GROBs für die Beschriftung
GOR	PICT mit Markierung GROB kombinieren
PICT STO	Kombiniertes GROB in PICT speichern
{ } PVIEW	PICT in den Stack ablegen
»	Programm beenden

Speichern Sie das Programm unter den Namen GRPR (GROB PRogram). Drücken Sie GRPR, um das Programm auszuführen. Die Ausgabe sieht wie folgt aus:



Ein Programm mit Plot- und Zeichenfunktionen

In diesem Abschnitt entwickeln wir ein Programm, mit dem der Mohr'sche Kreis für einen gegebenen zweidimensionalen Spannungszustand erzeugt, gezeichnet und beschriftet wird. Die Abbildung auf der linken Seite zeigt einen gegebenen Spannungszustand in zwei Dimensionen, wobei σ_{xx} und σ_{yy} normale Spannungen und $\tau_{xy} = \tau_{yx}$ Schubspannungen sind. Die Abbildung auf der rechten Seite zeigt den Spannungszustand, wenn das Element um einen Winkel von ϕ gedreht wird. In diesem Fall sind die normalen Spannungen σ'_{xx} und σ'_{yy} und die Schubspannungen τ'_{xy} und τ'_{yx}.

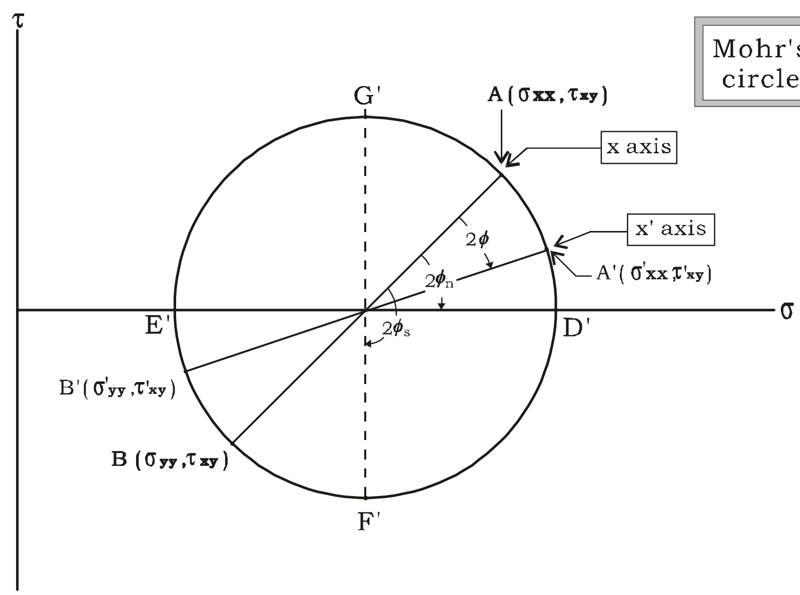


Das Verhältnis zwischen dem ursprünglichen Spannungszustand (σ_{xx}, σ_{yy}, τ_{xy}, τ_{yx}) und dem Spannungszustand nachdem die Achsen von f (σ'_{xx}, σ'_{yy}, τ'_{xy}, τ'_{yx})

gegen den Uhrzeigersinn gedreht wurden, kann mit der unten abgebildeten Figur grafisch dargestellt werden.

Für die Darstellung des Mohr'schen Kreises verwenden wir ein Kartesisches Koordinatensystem, in dem die x-Achse den normalen Spannungen (σ) und die y-Achse den Schubspannungen (τ) entspricht. Suchen Sie die Punkte A(σ_{xx},τ_{xy}) und B (σ_{yy}, τ_{xy}), und zeichnen Sie das Segment AB. Die Mitte des Kreises liegt im Punkt C, wo das Segment AB die Achse σ_n überschneidet. Beachten Sie, dass die Koordinaten des Punktes C ($\frac{1}{2}\cdot(\sigma_{yy} + \sigma_{xy})$, 0) sind. Wenn ein Kreis von Hand gezeichnet wird, können Sie mit dem Zirkel den Kreis zeichnen, da die Position der Kreismitte C und die zwei weiterer Punkte A und B bekannt ist.

Nehmen wir an, das AC-Segment ist die x-Achse in den ursprünglichen Spannungszustand. Wenn Sie den Spannungszustand eines Achsensystems x'-y' ermitteln wollen, das bezogen auf das ursprüngliche Achsensystem x-y um einen Winkel von ϕ gegen den Uhrzeigersinn gedreht ist, zeichnen Sie ein Segment A'B', dessen Mitte sich in C befindet und das um einen Winkel von 2ϕ bezogen auf AB in Uhrzeigersinn gedreht ist. Die Koordinaten von Punkt A' ergeben die Werte (σ'_{xx},τ'_{xy}) haben, während die von Punkt B' (σ'_{yy},τ'_{xy}) ergeben.



Der Spannungszustand bei dem die Schubspannung τ'_{xy} gleich Null ist, zu sehen im Segment D'E', erzeugt die sogenannten *Hauptspannungen*, σ^P_{xx} (bei Punkt D') und σ^P_{yy} (bei Punkt E') Die Hauptspannungen erhalten Sie, wenn Sie das Koordinatensystem x'-y' um einen Winkel von ϕ_n bezogen auf das Koordinatensystem x-y gegen den Uhrzeigersinn drehen. Im Mohr'schen Kreis beträgt der Winkel zwischen den Segmenten AC und D'C $2\phi_n$.

Der Spannungszustand, in dem die Schubspannung einen Spitzenwert τ'_{xy} erreicht, wird vom Segment F'G' gegeben. Unter diesen Umständen sind beide normale Spannungen gleich ($\sigma'_{xx} = \sigma'_{yy}$). Dieser Drehung entspricht der Winkel ϕ_s. Der Winkel zwischen den Segmenten AC und F'C in der Abbildung ist $2\phi_s$.

Modulare Programmierung

Für die Entwicklung des Programms, das den Mohr'schen Kreis in einem gegebenen Spannungszustand plottet, verwenden wir die modulare

Programmierung. Im Prinzip geht es hier darum, dass das Programm in mehrere Unterprogramme zerlegt wird, welche im Rechner als getrennte Variablen angelegt werden. Diese Unterprogramme werden anschließend von einem Hauptprogramm zusammengefasst. Das Hauptprogramm werden wir *MOHRCIRCL* benennen. Zuerst werden wir im HOME-Verzeichnis ein Unterverzeichnis mit dem Namen MOHRC anlegen und dann dieses Verzeichnis öffnen, um die Programme einzugeben.

Als nächstes müssen wir im Unterverzeichnis das Hauptprogramm und seine Unterprogramme anlegen.

Das Hauptprogramm MOHRCIRCL verwendet die folgenden Unterprogramme:

- INDAT : Fordert die Eingabe von σx, σy, τxy vom Benutzer, erstellt eine Liste σL = {σx, σy, τxy} als Ausgabe.
- CC&r : Verwendet σL als Eingabe, erzeugt σc = ½(σx+σy), r = Radius des Mohr'schen Kreises, φn = Winkel für Hauptspannungen, als Ausgabe.
- DAXES : Verwendet σc und r als Eingabe, ermittelt die Achsenbereiche und zeichnet die Achsen für die Grafik des Mohr'schen Kreises
- PCIRC : Verwendet σc, r und φn als Eingabe, zeichnet den Mohr'schen Kreis, indem ein parametrischer Plot erzeugt wird
- DDIAM : Verwendet σL als Eingabe, zeichnet das Segment AB (siehe Abbildung oben mit dem Mohr'schen Kreis), verbindet die Punkte aus den Eingangsdaten im Mohr'schen Kreis
- σLBL : Verwendet σL als Eingabe, setzt die Beschriftungen "σx" und "σy" zur Identifizierung der Punkte A und B.
- σAXS : Setzt die Beschriftungen "σ" und "τ" für die jeweiligen Achsen (x bzw. y).
- PTTL : Betitelt die Abbildung mit"Mohr's circle".

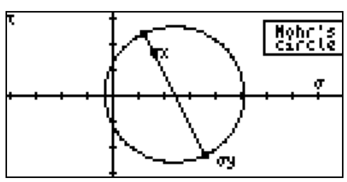
Die Programme finden Sie auf der mitgelieferten Diskette bzw. CD-ROM.

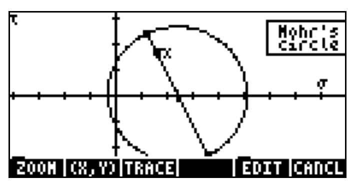
Programm ausführen

Wenn Sie die Programme in der oben aufgeführten Reihenfolge eingegeben haben, müssen im MOHRC-Unterverzeichnis die folgenden Variablen vorhanden sein: PTTL, σAXS, PLPNT, σLBL, PPTS, DDIAM. Wenn Sie NXT drücken, werden außerdem auch die folgenden Variablen angezeigt: PCIRC, DAXES, ATN2, CC&r, INDAT, MOHRC. Bevor die Variablen neu geordnet werden, führen Sie das Programm einmal aus, indem Sie die Funktionstaste MOHRC drücken. Verwenden Sie folgende:

MOHRC	Startet das Hauptprogramm MOHRCIRCL
2 5 ▼	Geben Sie σx = 25 ein
7 5 ▼	Geben Sie σy = 75 ein
5 0 ENTER	Geben Sie τxy = 50 ein und beenden Sie die Dateneingabe.

Jetzt beginnt das MOHRCIRCL-Programm die Unterprogramme aufzurufen, um die Figur zu erstellen. Seien Sie bitte geduldig. Der resultierende Mohr'sche Kreis wird wie auf dem Bild links aussehen.





Da diese Ansicht aus PICT durch die Funktion PVIEW erzeugt wird, erhalten wir keine weitere Informationen aus dem Plot, außer der Figur selbst. Um weitere Informationen aus dem Mohr'schen Kreis zu erhalten, müssen Sie das Programm mit ON beenden. Drücken Sie danach ◄, um die Inhalte von PICT in der Grafikumgebung wieder herzustellen. Der Mohr'sche Kreis wird diesmal wie auf dem Bild rechts aussehen (siehe oben).

Drücken Sie die Funktionstasten TRACE und (X,Y). Im unteren Bereich des Bildschirms sehen Sie den Wert von ϕ, der dem Punkt A(σx, τxy) entspricht, d.h., `φ = 0, (2.50E1, 5.00E1).`

Drücken Sie die rechte Pfeiltaste (▶), um den Wert von ϕ zu erhöhen und den entsprechenden Wert von (σ'_{xx}, τ'_{xy}) zu sehen. Zum Beispiel, für ϕ = 45° haben wir die Werte (σ'_{xx}, τ'_{xy}) = (1.00E2, 2.50E1) = (100, 25). Der Wert von σ'_{yy} befindet sich bei einem Winkel von 90° vorwärts, d.h. bei ϕ = 45 + 90 = 135°. Drücken Sie die Taste ▶, bis dieser Wert von ϕ erreicht wird, und Sie erhalten das folgende Ergebnis: (σ'_{yy}, τ'_{xy}) = (-1.00E-10,-2.5E1) = (0, 25).

Um die hauptsächlichen normalen Werte zu finden, drücken Sie ◀, bis der Cursor wieder an der Überschneidung des Kreises mit dem positiven Bereich der σ-Achse liegt. Für diesen Punkt finden wir die Werte ϕ = 59° und (σ'_{xx}, τ'_{xy}) = (1.06E2,-1.40E0) = (106, -1.40). Jetzt erwarten wir, dass der Wert von τ'_{xy} = 0 im Bereich der Hauptachsen liegt. Was aber geschieht, ist folgendes: Da wir die Auflösung der unabhängigen Variable auf $\Delta\phi$ = 1° beschränkt haben, verpassen wir den Punkt bei dem die Schubspannung Null ist. Wird ◀ noch einmal gedrückt, erhalten wir die Werte ϕ = 58° und (σ'_{xx}, τ'_{xy}) = (1.06E2,5.51E-1) = (106, 0.551). Diese Information sagt uns, dass irgendwo zwischen ϕ = 58° und ϕ = 59°, die Schubspannung τ'_{xy} Null wird.

Drücken Sie ON, um den eigentlichen Wert von ϕn zu ermitteln. Geben Sie danach die entsprechende Liste für die Werte {σx σy τxy}, in unserem Fall `{ 25 75 50 } [ENTER]`, ein.

Anschließend drücken Sie [illegible]. Das letzte Ergebnis in der Ausgabe, 58.2825255885°, ist der eigentliche Wert von ϕn.

Ein Programm für die Berechnung von Hauptspannungen

Der oben beschriebene Vorgang für die Berechnung von ϕn kann wie folgt programmiert werden:

Programm PRNST:

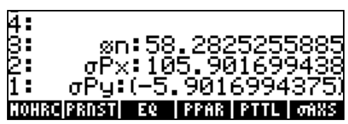
«	Das Programm PRNST (PRiNcipal Stresses – Hauptspannungen) starten
INDAT	Geben Sie die Daten wie beim MOHRCIRC-Programm ein

CC&r	Ermitteln Sie σc, r und fn, wie im MOHRCIRC-Programm
"ϕn" →TAG	Winkel für Hauptspannungen markieren
3 ROLLD	Markierten Winkel auf Ebene 3 verschieben
R→C DUP	σc und r in (σc, r) konvertieren, dann duplizieren
C→R + "σPx" →TAG	Hauptspannung σPx berechnen, dann markieren
SWAP C→R - "σPy" →TAG	Spannung σPy austauschen, berechnen, dann markieren.
»	Das Programm PRNST beenden

Um das Programm auszuführen, verwenden Sie:

VAR PRNST	Das Programm PRNST starten
2 5 ▼	Geben Sie σx = 25 ein
7 5 ▼	Geben Sie σy = 75 ein
5 0 ENTER	Geben Sie τxy = 50 ein und beenden Sie die Dateneingabe.

Die Lösung lautet:



Variablen im Unterverzeichnis sortieren

Beim ersten Lauf des Programms MOHRCIRCL wurden zwei neue Variablen – PPAR und EQ – erzeugt. Diese sind der Plot-Parameter und die Gleichungsvariablen, die zum Plotten des Kreises erforderlich sind. Wir schlagen vor, dass die Variablen aus dem Unterverzeichnis neu geordnet werden, so dass die Programme MOHRC und PRNST die ersten zwei Variablen unter den Funktionstasten sind. Dazu müssen Sie die Liste { MOHRCIRCL PRNST } wie folgt erstellen: VAR ← {} MOHRC PRNST ENTER

Sortieren Sie danach die Liste mit: ← PRG MEM DIR ORDER.

Nachdem der Funktionsaufruf ORDER ausgeführt wurde, drücken Sie [VAR]. Sie werden jetzt sehen, dass die Programme MOHRCIRCL und PRNST die ersten zwei Variablen im Menü sind, wie erwartet.

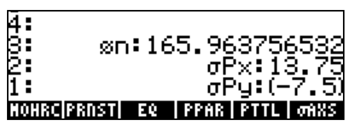
Ein zweites Beispiel zur Berechnungen des Mohr'schen Kreis

Ermitteln Sie die Hauptspannung für den Spannungszustand, der durch σ_{xx} = 12.5 kPa, σ_{yy} = -6.25 kPa und τ_{xy} = - 5.0 kPa definiert ist. Zeichnen Sie den Mohr'schen Kreis und ermitteln Sie aus der Figur die Werte von σ'_{xx}, σ'_{yy} und τ'_{xy} wenn der Winkel $\phi = 35°$.

Um die Hauptspannungen zu ermitteln, können wir das Programm [PRNST], wie nachfolgend gezeigt, anwenden:

[VAR] [PRNST]	Das Programm PRNST starten
[1][2][.][5][▼]	Geben Sie σx = 12.5 ein
[6][.][2][5][+/-][▼]	Geben Sie σy = -6.25 ein
5[+/-][ENTER]	Geben Sie τxy = -5 ein und beenden Sie die Dateneingabe.

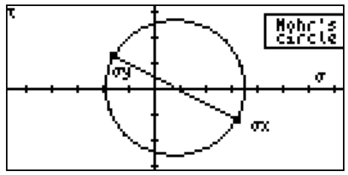
Die Lösung lautet:



Um den Mohr'schen Kreis zu zeichnen, verwenden Sie das Programm [MOHRC], wie folgt:

[VAR] [MOHRC]	Das Programm PRNST starten
[1][2][.][5][▼]	Geben Sie σx = 12.5 ein
[6][.][2][5][+/-][▼]	Geben Sie σy = -6.25 ein
5[+/-][ENTER]	Geben Sie τxy = -5 ein und beenden Sie die Dateneingabe.

Die Lösung lautet:



Für die Ermittlung der Spannungswerte, die einer Drehung von 35° des Winkels der gespannten Partikel entsprechen, gehen Sie wie folgt vor:

ON ◄	Bildschirm löschen, PICT auf dem Grafikbildschirm anzeigen
TRACE (X,Y).	Den Cursor über den Kreis bewegen, wo ϕ und (x,y) angezeigt werden.

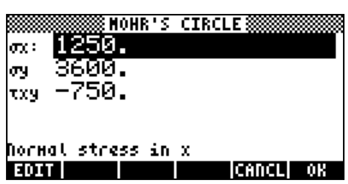
Drücken Sie zunächst ►, bis ϕ = 35 abgelesen wird. Die entsprechenden Koordinaten sind (1.63E0, -1.05E1), d.h. bei ϕ = 35°, σ'_{xx} = 1.63 kPa und σ'_{yy} = -10.5kPa.

Eingabemaske des Programms für den Mohr'schen Kreis

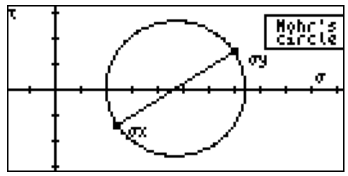
Eine weitere Möglichkeit die Daten einzugeben bietet sich, wenn das Unterprogramm INDAT durch das folgende Programm, das eine Eingabemaske aktiviert, ersetzt wird.

```
« "MOHR'S CIRCLE" { { "σx:" "Normal stress in x" 0 }
{ "σy:" "Normal stress in y" 0 } { "τxy:" "Shear stress"
0} }   { } { 1 1 1 } { 1 1 1 }  INFORM DROP »
```

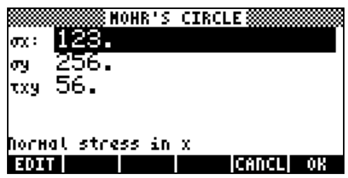
Wenn dieses Ersatzprogramm läuft, wird beim Ausführen von MOHRC die unten abgebildete Eingabemaske erzeugt:



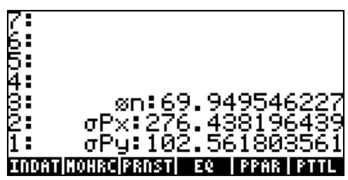
Drücken Sie OK, um den Programmlauf fortzusetzen: Das Ergebnis ist die folgende Figur:



Da das Programm INDAT auch für das Programm PRNST (PRiNcipal STresses) verwendet wird, wird beim Ausführen dieses Programms auch eine Eingabemaske verwendet, wie z.B.:



Nachdem OK gedrückt wird, erhalten Sie das folgende Ergebnis:

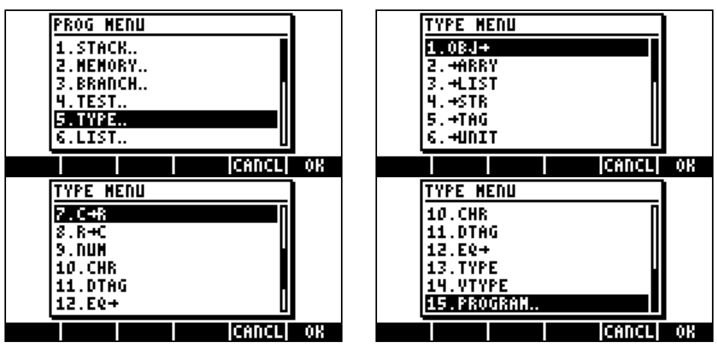


Kapitel 23
Zeichenketten

Zeichenketten sind Objekte, die zwischen Anführungszeichen eingeschlossen werden. Vom Rechner werden sie wie Text behandelt. So kann beispielsweise die Zeichenkette “SINE FUNCTION” in ein GROB (Grafikobjekt) umgewandelt werden, um eine Grafik zu benennen, oder um als Ausgabe eines Programmes verwendet zu werden. Jede Folge von Zeichen, die von einem Anwender als Eingabe für ein Programm gemacht wird, wird als Zeichenkette behandelt. Außerdem sind auch viele Programmausgaben Zeichenketten.

Zeichenketten-basierende Funktionen im Untermenü TYPE

Das Untermenü TYPE kann über das Menü PRG (Programmierung) aufgerufen werden, ← PRG . Im Menü TYPE stehen folgende Funktionen zur Verfügung:

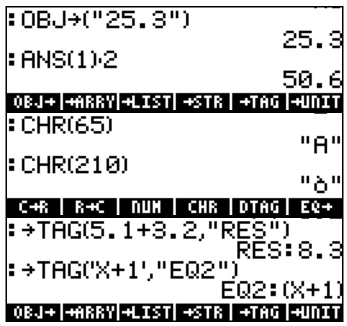


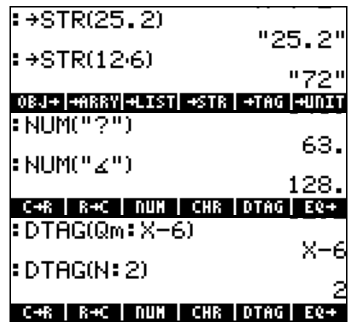
Zu den Funktionen im TYPE-Menü, die dazu dienen, Strings zu bearbeiten, gehören:

OBJ→	:	Konvertiert einen String in das repräsentierte Objekt
→STR	:	Konvertiert ein Objekt in einen entsprechenden String
→TAG	:	Markiert eine Menge
DTAG	:	Löscht die Markierung von einer markierten Menge (de-tag)
CHR	:	Erzeugt einen String mit einem Zeichen, entsprechend der Zahl die als Argument verwendet wird.

NUM : Gibt den Code des ersten Zeichen eines Strings zurück

Nachfolgend finden Sie Anwendungsbeispiele dieser Funktionen.





Verknüpfung von Zeichenketten

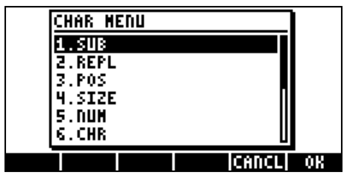
Strings können mit Hilfe des + Zeichens verknüpft (zusammen geführt) werden, beispielsweise:

Die Verknüpfung von Strings ist ein praktischer Weg, Ausgaben eines Programmes zu erzeugen. Verknüpfen Sie beispielsweise "YOU ARE " AGE + " YEAR OLD", wird, wenn in der Variablen AGE 25 gespeichert ist, "YOU ARE 25 YEAR OLD" ausgegeben.

Das Menü CHARS

Das Untermenü CHARS kann über das Menü PRG (Programmierung) aufgerufen werden, ← PRG .

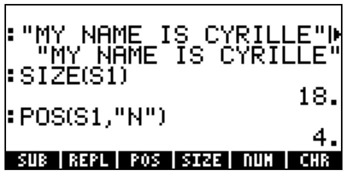
Im Menü CHARS stehen folgende Funktionen zur Verfügung:

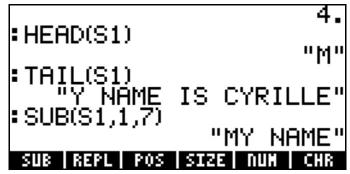


Die Operationen NUM, CHR, OBJ→ und →STR wurden bereits vorgestellt. Außerdem haben wir die Funktionen SUB und RPL in Bezug auf Grafiken kennengelernt. Die Funktionen SUB, REPL, POS, SIZE, HEAD und TAIL haben die folgenden Funktionen:

SIZE : Größe eines Sub-Strings in einer Zeichenkette (einschließlich der Leerstellen)
POS : Position des ersten Vorkommens eines Zeichens in einem String
HEAD : Ermittelt das erste Zeichen eines Strings
TAIL : Entfernt das erste Zeichen eines Strings
SUB : Wählt einen Sub-String mit gegebener Anfangs- und Endposition
REPL : Ersetzt Zeichen eines Strings durch einen Sub-String, beginnend an einer vorgegebenen Position
SREPL : Ersetzt einen Sub-String durch einen anderen Sub-String

Um zu sehen, wie diese Funktionen arbeiten, machen Sie folgende Übung: Speichern Sie den String "MY NAME IS CYRILLE" in die Variable S1. Wir verwenden diese Zeichenkette, um zu sehen, wie die Funktionen des CHARS-Menüs arbeiten:





Die Zeichenliste

Alle Zeichen, die auf dem Rechner zur Verfügung stehen, sind über die Tastenfolge [→] CHARS zugänglich. Wenn Sie ein Zeichen hervorheben, sagen wir beispielsweise das Zeichen Line Feed ↵ , sehen Sie unten links auf der Anzeige die Tastenfolge, die dieses Zeichen aufruft (→. In diesem Fall) und den numerischen Code des Zeichens (in diesem Fall 10).

Zeichen, die nicht definiert sind, erscheinen in der Zeichenliste als dunkle Rechtecke (▪) und unten auf der Anzeige wird (None) angezeigt, obwohl für alle ein numerischer Code existiert. Bei numerischen Zeichen wird der entsprechende numerische Code angezeigt.

Buchstaben zeigen den Code α (d.h., [ALPHA]) gefolgt von dem entsprechenden Buchstaben, wenn Sie beispielsweise M auswählen, wird unten links αM angezeigt, was bedeutet, dass [ALPHA][M] gedrückt werden müssen. Andererseits zeigt m die Tastenfolge α←M oder [ALPHA][←][M] .

Griechische Buchstaben wie beispielsweise σ zeigen den Code α→S oder [ALPHA][→][S] . Einigen Zeichen wie ρ ist keine Tastenfolge zugeordnet. Daher besteht für diese Zeichen nur die Möglichkeit, das entsprechende Zeichen in der Zeichenliste auszuwählen und dann durch Drücken von ECHO1 oder ECHO zu übertragen.

Drücken Sie ECHO1 , wird nur ein Zeichen in den Stack übertragen und der Rechner kehrt sofort in die normale Anzeige zurück. Verwenden Sie ECHO , um mehrere Zeichen in den Stack zu übertragen. Um zur normalen Anzeige zurückzukehren, drücken Sie ON.

In Anhang D finden Sie weitere Details zur Benutzung von Sonderzeichen. In Anhang G finden Sie Tastenkombinationen, mit denen Sie Sonderzeichen erstellen können.

Kapitel 24
Rechner-Objekte und Flags

Zahlen, Listen, Vektoren, Matrizen, Formeln usw. sind Rechner-Objekte. Sie werden in 30 verschiedene Typen unterteilt. Eine Beschreibung finden Sie weiter unten. Flags sind Variable die zur Steuerung von Eigenschaften des Rechner verwendet werden können. Flags wurden bereits in Kapitel 2 vorgestellt.

Beschreibung der Rechner-Objekte

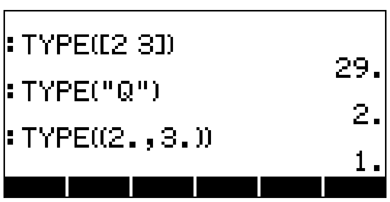
Der Rechner erkennt die folgenden Rechner-Objekte:

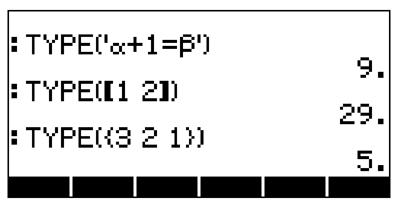
Nummer	Typ	Beispiel
0	Realzahl	-1.23E-5
1	Komplexe Zahl	(-1.2,2.3)
2	Zeichenkette	"Hello, world "
3	Reales Array	[[1 2][3 4]]
4	Komplexes Array	[[(1 2) (3 4)] [(5 6) (7 8)]
5	Liste	{3 1 'PI'}
6	Globale Namen	X
7	Lokale Namen	y
8	Programme	<< → a 'a^2' >>
9	algebraisches Objekt	'a^2+b^2'
10	Binär Integer	# A2F1E h
11	Grafik-Objekt	Graphic 131×64
12	Markiertes Objekt	R: 43.5
13	Einheiten-Objekt	3_m^2/s
14	XLIB-Namen	XLIB 342 8
15	Verzeichnis	DIR Σ END
16	Bibliothek	Library 1230"...
17	Sicherungs-Objekt	Backup MYDIR
18	Integrierte Funktionen	COS
19	Integrierte Befehle	CLEAR

Nummer	Typ	Beispiel
21	Erweiterte Realzahl	Long Real
22	Erweiterte komplexe Zahl	Long Complex
23	Verknüpfter Array	Linked Array
24	Zeichen-Objekt	Character
25	Code-Objekt	Code
26	Bibliotheks-Daten	Library Data
27	Externe Objekte	External
28	Integer	3423142
29	Externe Objekte	External
30	Externe Objekte	External

Funktion TYPE

Die Funktion dient dazu, den Typ eines Objekts zu bestimmen und kann entweder über das Menü PRG/TYPE () oder über den Befehlskatalog aufgerufen werden. Als Argument der Funktion wird das gesuchte Objekt eingesetzt. Die Funktion gibt den Typ des Objekts als eine der oben aufgeführten Nummern zurück.





Funktion VTYPE

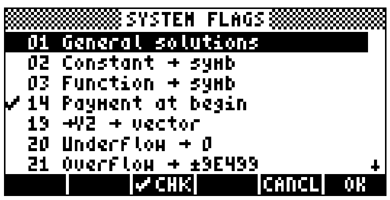
Diese Funktion arbeitet ähnlich wie die Funktion TYPE, wird aber bei Variablen verwendet. Sie gibt den Typ des Objekts, das in der Variablen gespeichert ist, zurück.

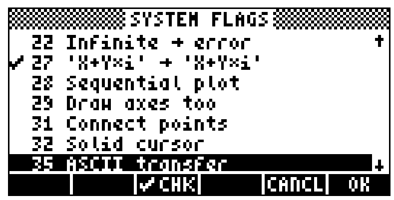
Rechner-Flags

Ein Flag ist eine Variable, die entweder gesetzt oder nicht gesetzt ist. Der Status des Flags beeinflusst das Verhalten des Rechners, bei einem System-Flag, einem Programm-Flag oder einem Anwender-Flag. Die einzelnen Typen werden nachfolgend beschrieben.

System-Flags

Auf System-Flags kann über MODE FLAGS zugegriffen werden. Drücken Sie die Taste mit dem nach unten weisenden Pfeil, um die Flags mit ihrer Nummer und einer Kurzbeschreibung anzuzeigen. Die ersten beiden Anzeigeseiten der System-Flags sehen sie unten:





Sie werden viele dieser Flags erkennen, da Sie im Menü MODES ein- oder ausgeschaltet werden (z.B. Flag 95 für algebraischer Modus, 103 für komplexer Modus usw.) In diesem Handbuch haben wir den Unterschied zwischen CHOOSE-Kästchen und SOFT-Menüs dargestellt. Sie werden gewählt, indem das System-Flag 117 gesetzt oder nicht gesetzt wird. Ein anderes Beispiel für System-Flags sind die Flags 60 und 61, die sich auf die Konstanten-Bibliothek auswirken (siehe CONLIB in Kapitel 3). Die Flags arbeiten auf folgende Weise:

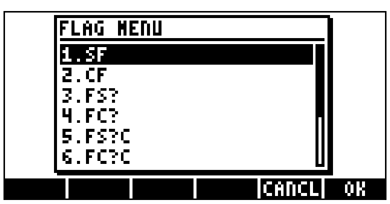
- Anwender-Flag 60: löschen (Vorgabe): SI-Einheiten, setzen: ENGL-Einheiten
- Anwender-Flag 61: löschen (Vorgabe): Einheiten verwenden, setzen: nur Werte
-

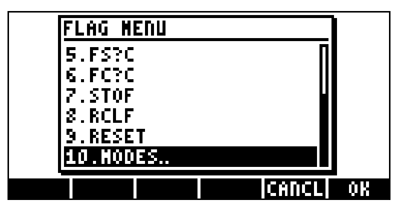
Funktionen zum Setzen und Ändern von Flags

Diese Funktionen können dazu verwendet werden, Anwender- oder System-Flags zu setzen, zu löschen oder ihren Status zu prüfen. Auf System-Flags wird mit diesen Funktionen über negative Integerzahlen zugegriffen. So wird auf das System-Flag 117 über -117 zugegriffen. Für Anwender-Flags werden hierzu positive Integerzahlen verwendet. Es ist wichtig, dass Anwender-Flags nur in der Programmierung Anwendung finden, um den Programmfluss zu steuern.

Funktionen zur Manipulation der Rechner-Flags finden Sie im Menü PRG/MODES/FLAG Das PRG-Menü wird die die Tastenfolge ← PRG aufgerufen Die folgenden Anzeigefenster zeigen, wie man (mit dem System Flag 117 auf CHOOSE Kästchen eingestellt) zu dem Menü FLAG gelangt:

Im Menü FLAG sind die folgenden Funktionen enthalten:





Die Funktionen arbeiten wie folgt:

SF Flag setzen
CF Flag löschen
FS? Gibt 1 zurück, wenn das Flag gesetzt ist, 0, wenn es nicht gesetzt ist
FC? Gibt 1 zurück, wenn das Flag leer ist (nicht gesetzt), 0, wenn es gesetzt ist

FS?C Testet Flag wie FS und löscht es dann
FC?C Testet Flag wie FC und löscht es dann
STOF Speichert neue System-Flag-Einstellungen
RCLF Lädt die bestehenden Flag-Einstellungen
RESET Setzt die gegenwärtigen Feldwerte zurück (kann zum Zurücksetzen eines Flag verwendet werden)

Anwender-Flags

Zu Programmierzwecken stehen dem Anwender die Flags von 1 bis 256 zur Verfügung. Sie beeinflussen die Funktion des Rechners nicht.

Kapitel 25
Datum- und Zeit-Funktionen

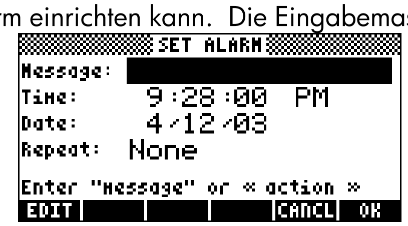
In diesem Kapitel demonstrieren wir einige Funktionen und Berechnungen mit Zeiten und Daten.

Das Menü TIME

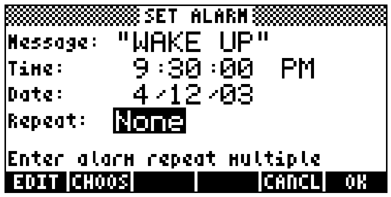
Das Menü TIME wird über die Tastenfolge ⇨ TIME (die Taste 9) aufgerufen und bietet die nachfolgend beschriebenen Funktionen:

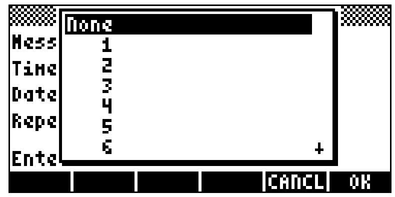
Alarm einrichten

Die Option *2. Set alarm..* bietet eine Eingabemaske, mit deren Hilfe der Anwender einen Alarm einrichten kann. Die Eingabemaske sieht wie folgt aus:



Das Eingabefeld Message bietet Ihnen die Möglichkeit einen Text einzugeben, der den Alarm beschreibt. In das Eingabefeld Time können Sie die gewünschte Zeit eingeben. Mit dem Feld Datum stellen Sie das Datum für den Alarm ein (oder bei wiederholtem Alarm den Tag für die erste Aktivierung) Sie können beispielsweise den folgenden Alarm einrichten: Die linke Abbildung zeigt einen Alarm ohne Wiederholung. Die rechte Abbildung zeigt die Optionen für Wiederholungen, wenn CHOOS gedrückt wurde. Nach dem Drücken von OK ist der Alarm eingestellt.





Alarme suchen

Mit der Option *1. Browse alarms…* im Menü TIME können Sie durch Ihre gegenwärtigen Alarme blättern. Wenn Sie beispielsweise den obigen Alarm eingegeben haben, zeigt Ihnen die Option die folgende Anzeige:

Diese Anzeige enthält vier Funktionstasten:

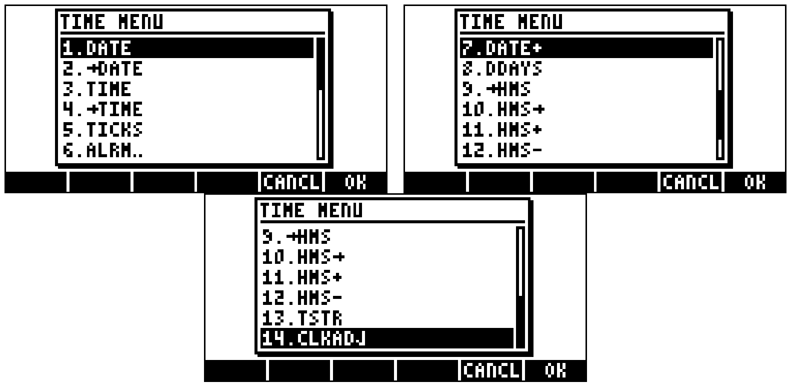
EDIT : Editieren des Alarms über die Eingabemaske
NEW : Eingabe eines neuen Alarms
PURG : Löschen eines Alarms
OK : Rückkehr zur normalen Anzeige

Datum und Uhrzeit einstellen

Die Option *3. Set time, date..* bietet eine Eingabemaske, mit deren Hilfe der Anwender die momentane Zeit und das Datum einstellen kann. Details wurden bereits in Kapitel 1 vorgestellt.

TIME-Funktionen

Die Option *4. Tools…* stellt eine Reihe von Funktionen für Uhren-Funktionen und Kalkulationen mit Zeit und Datum zur Verfügung. In der folgenden Abbildung sehen Sie die einzelnen TIME-Funktionen:

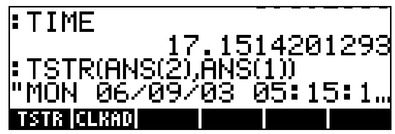


Die Funktionen arbeiten wie folgt:

DATE	: Stellt das gegenwärtige Datum in den Stack
→DATE	: Stellt das Systemdatum auf den eingegebenen Wert
TIME	: Stellt die momentane Zeit im Format HH.MMSS in den Stack.
→TIME	: Stellt die momentane Zeit im Format HH.MMSS auf den eingegebenen Wert ein.
TICKS	: Stellt die Systemzeit als binäre Integerzahl in Zeit-Ticks mit 1 Tick = 1/8192 Sekunden zur Verfügung
ALRM..	: Untermenü mit Funktionen zur Manipulation von Alarmen (Beschreibung folgt weiter unten)
DATE+	: Addiert oder Subtrahiert eine Anzahl an Tagen zu einem Datum
DDAYS(x,y)	: Gibt die Anzahl von Tagen zwischen den Daten x und y zurück
→HMS	: Konvertiert die Zeit aus einer Dezimalzahl in das Format HH.MMSS
HMS→	: Konvertiert die Zeit aus dem Format HH.MMSS in eine Dezimalzahl
HMS+	: Addiert zwei Zeiten im Format HH.MMSS
HMS-	: Subtrahiert zwei Zeiten im Format HH.MMSS
TSTR(time, date)	: Konvertiert Zeit, Datum in eine Zeichenkette
CLKADJ(x)	: Addiert x Ticks auf die Systemzeit (1 Tick = 1/8192 s)

Die Funktionen →DATE, →TIME, CLKADJ werden zur Einstellung der Zeit und des Datums verwendet. Für diese Funktionen werden keine Beispiele angeführt.

Nachfolgend finden Sie Beispiele für die Funktionen DATE, TIME und TSTR:

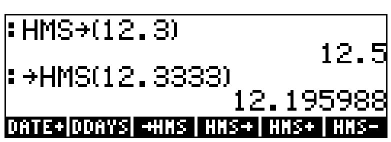


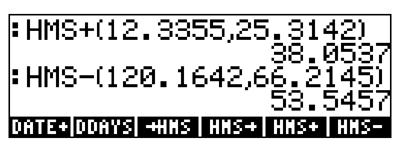
Berechnungen mit Daten

Zur Berechnung mit Daten verwenden Sie die Funktionen DATE+ und DDAYS. Nachfolgend finden Sie ein Beispiel für diese Funktionen zusammen mit einem Beispiel der Funktion TICKS:

Berechnungen mit Zeit

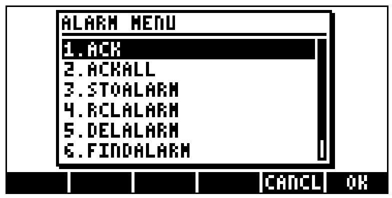
Die Funktionen →HMS, HMS→, HMS+ und HMS- werden zur Manipulation von Werten im Format HH.MMSS verwendet. Hierbei handelt es sich um das gleiche Format wie bei der Berechnung von Winkeln in Grad, Minuten und Sekunden. Daher sind diese Funktionen nicht nur für Zeit- sondern auch für Winkelberechnungen von Vorteil. Nachfolgend einige Beispiele:





Alarm-Funktionen

In dem Menü TIME/Tools.../ALARM... finden Sie folgende Funktionen:



Die Operation dieser Funktionen wird als nächstes erläutert:

ACK : Bestätigung eines überfälligen Alarms
ACKALL: Bestätigung aller überfälligen Alarme
STOALARM(x) : Speichern des Alarm (x) in die Systemalarmliste
RCLALARM(x) : Laden des Alarm (x) aus der Systemalarmliste
DELALARM(x) : Löschen des Alarm (x) aus der Systemalarmliste
FINDALARM(x) : Gibt den ersten überfälligen Alarm nach der spezifizierten Zeit zurück

Das Argument x in der Funktion STOALARM besteht aus einer Liste, die ein Referenzdatum (mm.ddyyy), eine Tageszeit im 24-Stunden-Format (hh.mm), eine Zeichenkette mit dem Alarmtext und die Anzahl der Wiederholungen enthält. Beispielsweise `STOALARM({6.092003,18.25,"Test",0}`. Das Argument x in allen anderen Alarm-Funktionen ist eine positive Integerzahl, die sich auf den Alarm bezieht, der aufgerufen, gelöscht oder gefunden werden soll.

Da Alarmeinstellungen leicht über das Menü TIME (siehe oben) erfolgen können, beziehen sich die Alarm-Funktionen eher auf die Programmierung des Rechners.

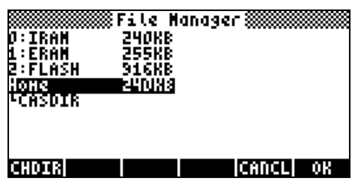
Kapitel 26

Speicherverwaltung

In dem Kapitel 2 des Benutzerhandbuches haben wir Sie mit der Erstellung und Verwaltung von Variablen und Verzeichnissen vertraut gemacht. In diesem Kapitel werden wir die Speicherverwaltung des Rechners in Speicherpartitionen und die Technik der Datensicherung diskutieren.

Speicheraufbau

Der Rechner besitz eine Speicherkapazität von 2,5 MB. 1 MB wird durch das Betriebssystem belegt (Systemspeicher) und 1,5 MB werden für Berechnungen und der Speicherung von Daten verwendet (Arbeitsspeicher). Anwender können auf den Systemspeicherbereich nicht zugreifen. Um zu sehen, wie der Arbeitsspeicher partitioniert ist, benutzen Sie die Funktion FILES (← FILES). Nachfolgend sehen Sie ein mögliches Ergebnis:



Neben dem Speicher der dem HOME-Verzeichnis (siehe Kapitel 2 im Benutzerhandbuch) zugeordnet ist, sehen Sie drei weitere Speicherbereiche. Zu diesen Speicherbereichen, genannt Ports, gehören:

- Port 0, mit der Bezeichnung IRAM
- Port 1, mit der Bezeichnung ERAM
- Port 2, mit der Bezeichnung FLASH

Port 0 und das HOME-Verzeichnis teilen sich den gleichen Speicherbereich. Je mehr Daten also im HOME-Verzeichnis gespeichert werden, desto weniger Platz ist für Port 0 verfügbar. Der Gesamtspeicherplatz für Port 0 und HOME beträgt 241 KB.

Port 1 (ERAM) kann bis zu 255 KB an Daten enthalten. Port 1 bildet zusammen mit Port 0 und dem HOME-Verzeichnis das RAM (Random Access Memory) des Rechners. Das RAM muss ständig über die Batterien mit Strom versorgt werden.Um Datenverlust des RAMs zu verhindern, ist eine Pufferbatterie CR2032 integriert. Sie zusätzliche Details am Ende dieses Kapitels.

Port 2 gehört zum Flash-ROM (Read only Memory) des Rechners und benötigt keine Stromversorgung. Daher hat es für das Flash-ROM keine Bedeutung, wenn die Batterien entfernt werden. Port 2 kann bis zu 1085 KB Daten speichern.

Das HOME Verzeichnis

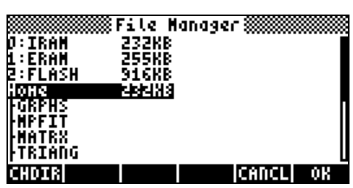
Wenn Sie den Rechner benutzen, erstellen Sie Variablen, die Sie speichern wollen, Zwischen- und Endergebnisse. Einige Rechnerfunktionen, wie Grafik und Statistik, erzeugen ihre eigenen Variablen, um Daten zu speichern. Diese Variablen befinden sich dann im HOME-Verzeichnis oder eines der Unterverzeichnisse. Genaueres über die Manipulation von Variablen und Verzeichnissen können Sie in Kapitel 2 des Benutzerhandbuches nachlesen.

Speicher-Port

Anders als das HOME-Verzeichnis kann der Speicher-Port in Unterverzeichnisse aufgeteilt werden. Es kann nur Sicherungsobjekte oder Bibliotheken enthalten. Diese Objektarten werden nachfolgend beschrieben.

Prüfen von Objekten im Speicher

Um zu sehen, welche Objekte im Speicher abgelegt sind, benutzen Sie die Funktion FILES (← FILES). Die Anzeige zeigt das HOME-Verzeichnis mit wenigsten vier Verzeichnissen, GRPHS, MPFIT, MATRX und TRIANG.



Zusätzliche Verzeichnisse können angezeigt werden, indem Sie den Cursor im Verzeichnisbaum nach unten bewegen. Oder bewegen Sie den Cursor nach oben, um sich einen Speicher-Port anzeigen zu lassen. Wurde ein Verzeichnis, ein Unterverzeichnis oder ein Port ausgewählt, drücken Sie OK , um den Inhalt des ausgewählten Objekts anzuschauen. Zusätzlich können Sie auf einen Port zugreifen, indem Sie im Menü LIB (→ LIB , zusammen mit der Taste 2 verwenden. Hierdurch erscheint folgende Anzeige:

Ist eine Bibliothek aktiv, wird sie hier angezeigt. Eine solche Bibliothek ist HP49D (Demobibliothek), die oben angezeigt wird. Wenn Sie die entsprechende Funktionstaste (F1) drücken, wird die Bibliothek aktiviert. Drücken Sie die Port-Funktionstasten, wird der Speicher-Port geöffnet. Zusätzliche Informationen zu Bibliotheken finden Sie weiter unten.

Sicherungs-Objekte

Sicherungs-Objekte dienen dazu, Daten von Ihrem HOME-Verzeichnis in einen Speicher-Port zu schreiben. So kann der Inhalt dieser Objekte für zukünftige Nutzung gesichert werden. Sicherungs-Objekte haben die folgenden Merkmale:

- Sie können nur in Speicher-Ports existieren (das bedeutet, dass Sie keine Objekte im HOME-Verzeichnis sichern können, obwohl Sie soviele Kopien erzeugen können, wie Sie wollen)
- Der Inhalt von Sicherungs-Objekten kann nicht modifiziert werden (Sie können jedoch das Objekt zurück in das HOME-Verzeichnis kopieren, dort modifizieren und es dann wieder sichern)
- Sie können ein einzelnes Objekt oder ein ganzes Verzeichnis als Sicherungs-Objekt speichern. Sie können aber kein Sicherungs-Objekt aus mehreren ausgewählten Objekten eines Verzeichnisses erstellen.

Wenn Sie ein Sicherungs-Objekt erstellen, erzeugt der Rechner eine Prüfsumme (*checksum* oder CRC) basierend auf den Binärdaten des Objekts.

Dieser Wert wird zusammen mit dem Sicherungs-Objekt gespeichert und dient dem Rechner dazu, die Integrität dieser Objekte zu überwachen. Wenn Sie ein Sicherungs-Objekt in das HOME-Verzeichnis zurückschreiben, errechnet der Rechner wieder die Prüfsumme und vergleicht diese mit dem gespeicherten Wert. Werden Unterschiede festgestellt, warnt Sie der Rechner, das die zurückgeschriebenen Daten beschädigt sein können.

Sichern von Objekten im Speicher-Port

Sicherungs-Objekte werden genau so vom Arbeitsspeicher in einen Speicher-Port geschrieben, wie Variablen von einem Unterverzeichnis in das andere kopiert werden (siehe Kapitel 2 des Bedienerhandbuchs). So können Sie beispielsweise den Dateimanager (← FILES) dazu benutzen, Sicherungs-Objekte zu kopieren und zu löschen, wie Sie es auch mit normalen Rechner-Objekten machen. Wie im folgenden beschrieben, gibt es zusätzlich spezielle Befehle zur Manipulation von Sicherungs-Objekten.

HOME sichern und neu laden

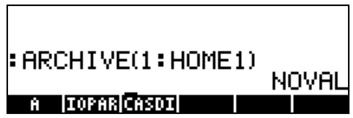
Sie können den Inhalt des gesamten HOME-Verzeichnisses in einem einzigen Sicherungs-Objekt speichern. Dieses Objekt enthält dann alle Variablen, Tastenzuordnungen und Alarme, die zu dem Zeitpunkt im HOME-Verzeichnis definiert waren. Sie können auch den Inhalt Ihres HOME-Verzeichnisses aus einem Sicherungs-Objekt, das vorher gespeichert wurde, wiederherstellen. Die Anleitung hierzu folgt weiter unten.

Sichern des HOME-Verzeichnisses

Um das momentane HOME-Verzeichnis im algebraischen Modus zu sichern, geben Sie folgenden Befehl ein:

ARCHIVE(:Port_Number: Backup_Name)

Hier sind Port_Number 0, 1, 2 (oder 3, wenn eine SD-Speicherkarte zur Verfügung steht – siehe unten) und Backup_Name ist der Name, dem Sie dem Sicherungs-Objekt für den Inhalt von HOME zuordnen wollen. Der _ : Container wird durch die Tastenfolge ← :: eingegeben. Um beispielsweise den Inhalt von HOME in HOME1 im Port 1 zu sichern, verwende Sie:



Um das momentane HOME-Verzeichnis im RPN-Modus zu sichern, geben Sie folgenden Befehl ein:

: Port_Number : Backup_Name ENTER ARCHIVE

Wiederherstellen des HOME Verzeichnisses
Um das momentane HOME-Verzeichnis im algebraischen Modus wiederherzustellen, geben Sie folgenden Befehl ein:

RESTORE(: Port_Number : Backup_Name)

Um das HOME-Verzeichnis aus Ihrem Sicherungs-Objekt HOME 1 wiederherzustellen, geben Sie ein:

RESTORE(:1:HOME1)

Im RPN-Modus geben Sie ein:

: Port_Number : Backup_Name ENTER RESTORE

Anmerkung: Wenn Sie ein HOME-Verzeichnis wiederherstellen, passieren zwei Dinge:

- Das gesicherte Verzeichnis überschreibt das momentane HOME-Verzeichnis Alle Daten des momentanen HOME-Verzeichnisses, die nicht gesichert wurden, gehen verloren.
- Der Rechner startet neu. Die Inhalte der History oder des Stacks gehen verloren.

Speichern, Löschen und Wiederherstellen von Sicherungs-Objekten

Um ein Sicherungs-Objekt zu speichern, wählen Sie eine der folgenden Möglichkeiten:

- Wählen Sie den Dateimanager (← FILES) , um das Objekt in den Port zu kopieren. Nach dieser Methode hat das Sicherungs-Objekt den gleichen Namen wie das Original.
- Kopieren Sie das Objekt mit STO in den Port. Um beispielsweise im algebraischen Modus die Variable A in ein Sicherungs-Objekt mit dem Namen AA im Port 1 zu speichern, geben Sie nachstehende Tastenfolge ein:

  ~~~ STO▸ ← :: 1 ▸ ALPHA A ALPHA A ENTER
-
- Um eine Sicherung Ihres HOME-Verzeichnisses zu erstellen, verwenden Sie den Befehl ARCHIVE (siehe oben).

Um ein Sicherungs-Objekt in einem Port zu löschen:

- Verwenden Sie den Dateimanager (← FILES) genau so wie Sie es vom Löschen einer Variablen im HOME-Verzeichnis kennen (siehe Kapitel 2 des Bedienerhandnuches), um das Objekt zu löschen.
- Verwenden Sie den Befehl PURGE wie folgt:
  Im algebraischen Modus: PURGE(: Port_Number : Backup_Name)
  Im RPN-Modus geben Sie ein: : Port_Number : Backup_Name PURGE

Um ein Objekt wiederherzustellen:

- Wählen Sie den Dateimanager (← FILES) , um das Objekt von dem Speicher-Port in das HOME-Verzeichnis zu kopieren.
- Nachdem das Sicherungs-Objekt wiederhergestellt wurde, für der Rechner eine Integritätsprüfung durch und kalkuliert den CRC-Wert. Eine Differenz zwischen der errechneten und der gespeicherten Prüfsumme resultiert in einer Fehlermeldung über zerstörte Daten.
-
  ~~~

Daten aus Sicherungs-Objekten verwenden

Obwohl Sie den Inhalt eines Sicherungs-Objekts nicht direkt modifizieren können, können Sie den Inhalt aber für Rechenoperationen verwenden. Sie können beispielsweise ein Programm, das Sie als Sicherungs-Objekt gespeichert haben, starten oder in einem Sicherungs-Objekt gesicherte Daten für ein Programm verwenden. Um gesicherte Programme oder Daten zu verwenden, benutzen Sie den Dateimanager (← FILES) , um das Sicherungs-Objekt in die Anzeige zu kopieren. Alternativ können Sie ein gesichertes Programm mit dem Befehl EVAL starten oder gesicherte Daten mit der RCL wiederherstellen. Gehen Sie wie folgt vor:

- Im algebraischen Modus:
  - Um ein Sicherungs-Objekt auszuwerten, geben Sie ein:
    EVAL(Argument(e), : Port_Number : Backup_Name)
  - Um ein Sicherungs-Objekt auszuwerten, geben Sie ein:
    Um ein Sicherungs-Objekt in die Befehlszeile zu laden, geben Sie ein:
    RCL(: Port_Number : Backup_Name)

- Im RPN-Modus:
  - Um ein Sicherungs-Objekt auszuwerten, geben Sie ein:
    Argument(e) ENTER : Port_Number : Backup_Name EVAL
  - Um ein Sicherungs-Objekt auszuwerten, geben Sie ein:
    : Port_Number : Backup_Name ENTER RCL

Verwenden von SD-Karten

Der Rechner bietet eine Speicherkartenschnittstelle, in welche Sie eine SD-Flash-Karte einschieben können, um entweder die Rechenobjekte zu sichern oder um Objekte von anderen Quellen herunter zu laden. Stecken Sie die Karte mit der Beschriftung nach unten ein. Die SD-Karte wird im Rechner als Schnittstelle (Port) Nummer 3 angezeigt.

Das Laden eines Objektes von der SD-Karte ist ähnlich, wie das Laden der Objekte von den Ports 0, 1 oder 2. Wenn Sie aber die Funktion LIB (→ LIB) benutzen, wird Port 3 im Menü angezeigt. Die Dateien einer SD-

Karte können nur mit dem Filer oder dem Dateimanager (← FILES) verwaltet werden. Wenn Sie den Filer starten, erscheint folgender Baumstruktur:

```
0: IRAM
1: ERAM
2: FLASH
3: SD
HOME
|-- sub-directories
```

Begeben Sie sich in die Baumstruktur der SD-Karte, werden alle Objekte als Backup-Objekte angezeigt. Deshalb ist es nicht möglich zu sagen, welches Objekt von welchem Typs ist, wenn man dessen Namen im Filer sieht. Der Filer unterstützt keine langen Namen Daher müssen alle Namen, wie bei DOS, im Format 8.3 vergeben werden, d.h., Namen mit maximal 8 Zeichen und einem Anhang von 3 Zeichen.

Als Alternative zu Datei-Manager Operationen, können Sie die Funktionen STO und RCL zum Speichern und Laden der Objekte von der SD-Karte, wie unten gezeigt, benutzen. Um Sicherungs-Objekte auf der SD-Speicherkarte zu löschen, können Sie auch den Befehl PURGE verwenden. Lange Namen können mit den Befehlen STO, RCL und PURGE verwendet werden.

Speichern von Objekten auf der SD-Karte

Objekte können nur im Hauptverzeichnis der SD-Karte gespeichert werden, d.h. auf Port 3 kann kein Unterverzeichnis angelegt werden (Diese Funktion wird wahrscheinlich in einem späteren Upgrade des Flash-ROMs verbessert. Benutzen Sie die Funktion STO, um ein Objekt zu speichern, wie folgt:

- Im algebraischen Modus:
  Geben Sie das Objekt ein, drücken Sie STO▶, geben Sie den Namen des zu speichernden Objektes, unter Verwendung des Port 3 ein (z.B. :3:VAR1), danach drücken Sie ENTER.
- Im RPN-Modus:
  Geben Sie das Objekt ein, dann den Namen des zu speichernden Objektes, unter Verwendung des Port 3 (z.B. :3:VAR1) und drücken dann STO▶.

Laden eines Objektes von der SD-Karte

Um ein Objekt von der SD-Karte in die Anzeige des Rechners zu laden, benutzen Sie die Funktion RCL wie folgt:

- Im algebraischen Modus:
  Drücken Sie ← RCL , geben Sie den Namen des zu speichernden Objektes, unter Verwendung des Port 3 ein (z.B. :3:VAR1) und drücken dann ENTER .
- Im RPN-Modus:
  Geben Sie den Namen des zu speichernden Objektes, unter Verwendung des Port 3 ein (z.B. :3:VAR1) und drücken dann ← RCL .

Mit dem Befehl RCL ist es möglich, Variablen durch Angabe eines spezifischen Pfades im Befehl wiederherzustellen, z.B. im RPN-Modus: :3: {path} ENTER RCL. Der Pfad ist, wie bei einem DOS-Laufwerk, eine Reihe von Verzeichnisnamen, die die Position einer Variablen in einem Verzeichnispfad definieren. Einige Variablen innerhalb eines Sicherungs-Objekts können jedoch nicht über einen Pfad wiederhergestellt werden. In diesem Fall muss das gesamte Sicherungs-Objekt (z.B. ein Verzeichnis) wiederhergestellt werden, damit auf einzelne Variablen zugegriffen werden kann.

Löschen eines Objektes von der SD-Karte

Um ein Objekt von der SD-Karte zu löschen und in die Anzeige des Rechners zu laden, benutzen Sie die Funktion PURGE wie folgt:

- Im algebraischen Modus:
  Drücken Sie TOOL PURGE, geben Sie den Namen des gesicherten Objekts und Port 3 ein (e.g., :3:VAR1) und drücken ENTER .
- Im RPN-Modus:
  Geben Sie den Namen des gesicherten Objekts und Port 3 ein (e.g., :3:VAR1) und drücken TOOL PURGE.

Verwendung von Bibliotheken

Bibliotheken sind vom Anwender erstellte Programme in binärer Sprache, die in den Rechner geladen werde können und aus einem Unterverzeichnis des

HOME-Verzeichnisses aufgerufen werden. Sie können als reguläre Variablen in den Rechner geladen, installiert und an das HOME-Verzeichnis angehängt werden.

Installation und Anhängen von Bibliotheken

Um eine Bibliothek zu installieren, listen Sie den Bibliotheksinhalt aus dem Stack auf (mit der Funktionstaste ⮫ oder der Funktion RCL) und speichern sie in Port 0 oder 1. Um beispielsweise eine Bibliothek in einen Port zu installieren, geben Sie ein:

- Im algebraischen Modus: STO(Library_variable, port_number)
- Im RPN-Modus: Library_variable ENTER port_number STO▸

Nach der Installation der Bibliothek im Speicher-Port müssen Sie die Bibliothek an das HOME-Verzeichnis anhängen.   Dies kann geschehen, indem Sie den Rechner neu starten (Rechner aus- und wieder einschalten) oder indem Sie gleichzeitig ON F3 drücken. Jetzt kann die Bibliothek verwendet werden. Um das Menü der Bibliotheksaktivierung anzuzeigen, verwenden Sie das Menü LIB (⮫ LIB).   Der Bibliotheksname wird nun in dem Menü aufgeführt.

Bibliotheksnummern

Wenn Sie das Menü LIB (⮫ LIB) verwenden und die Funktionstaste des Port 0 oder 1 drücken, sehen Sie eine Auflistung der Bibliotheksnummern. Jeder Bibliothek ist eine vierstellige Nummer zugeordnet. Diese Nummern werden von der Funktion, die die Bibliotheken erzeugt, zugeordnet und werden beim Löschen einer Bibliothek verwendet.

Bibliothek löschen

Um eine Bibliothek in einem Port zu löschen, geben Sie folgendes ein:

- Im algebraischen Modus: PURGE(:port_number: lib_number)
- Im RPN-Modus: port_number : lib_number PURGE

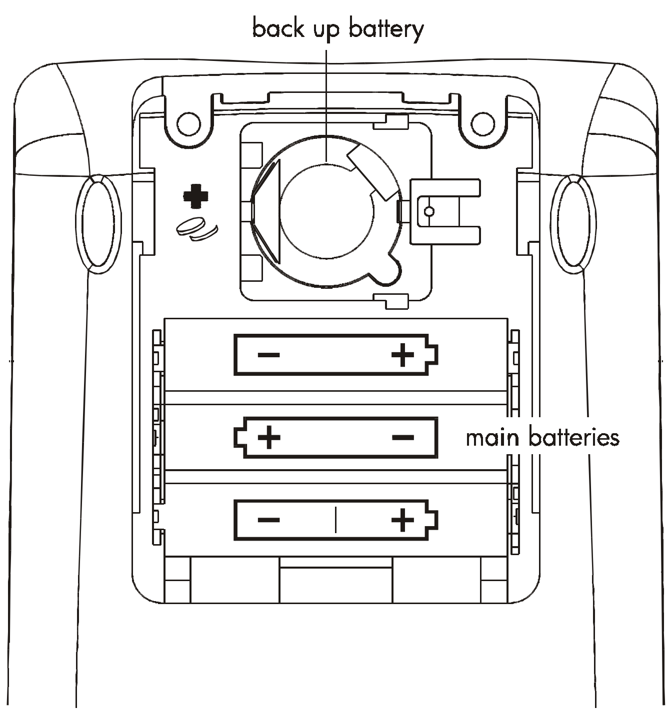
lib_number ist die oben beschriebene Bibliotheksnummer.

Bibliothek erstellen

Eine Bibliothek kann mit Hilfe der Programmiersprache Assembler, mit der Systemsprache RPL oder einer Matrix, die Bibliotheken erzeugt, wie LBMKR erstellt werden. Das letzte Programm ist im Internet verfügbar (siehe beispielsweise http://www.hpcalc.org). Details zur Programmierung in Assembler oder RPL liegen außerhalb des Umfangs dieses Handbuchs. Weitere Informationen über dieses Gebiet finden Sie im Internet.

Pufferbatterie

Um den flüchtigen Speicher bei dem Austausch der Batterien vor Datenverlust zu schützen, ist der Rechner mit einer Pufferbatterie CR2032 ausgestattet. Die Batterie sollte alle 5 Jahre ausgetauscht werden. Wenn die Batterie in Ihrer Leistung nachlässt, erscheint eine entsprechende Meldung auf der Anzeige. Die nachfolgende Darstellung zeigt Ihnen die Position der Batterie oben im Gehäuse auf der Rückseite des Rechners.



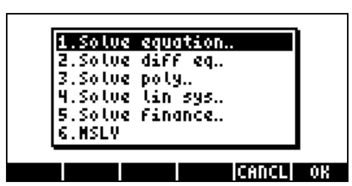
Anhang A
Benutzung von Eingabeformularen

Dieses Beispiel, bei dem Zeit und Datum gesetzt werden, veranschaulicht den Gebrauch von Eingabeformularen mit dem Rechner. Einige Grundregeln:

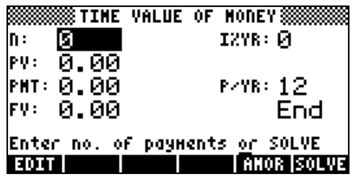
- Benutzen sie die Pfeitasten (◀ ▶ ▼ ▲) um von einem Feld des Eingabeformulars zum nächsten zu springen.
- Drücken Sie irgendeine der CHOOS Soft-Menü Tasten, um die zur Verfügung stehenden Optionen für ein gegebenes Feld des Eingabeformulars zu sehen.
- Benutzen Sie die Pfeiltasten (◀ ▶ ▼ ▲) um die gewünschte Option für ein gegebenes Feld auszuwählen und drücken Sie die OK (F6) Tasten des Soft-Menüs um die Auswahl zu aktivieren.
- In einigen Fällen muss eine Check-Marke benutzt werden, um eine Option in einem Eingabeformular auszuwählen. In einem solchen Fall benutzen Sie die CHK Taste des Soft-Menüs um die Check-Marke ein und aus zu schalten.
- Drücken Sie die CANCL Taste des soft-Menüs um ein Eingabeformular zu schliessen und zum Stack Display zurück zu kehren. Sie können auch die ENTER Taste oder die ON Taste drücken, um das Eingabeformular zu schliessen.

Beispiel – Benutzung von Eingabeformularen im NUM.SLV Menü

Bevor wir diese Themen im Detail besprechen, zeigen wir Ihnen einige der Eigenschaften von Eingabeformularen anhand der Finanzkalkulations-Anwendung des Numerical Solvers. Starten Sie den Nemerical Solver durch drücken von ↱ NUM.SLV (zusammen mit der 7 Taste). . Es erscheint eine Auswahl-Box, die folgende Optionen enthält:



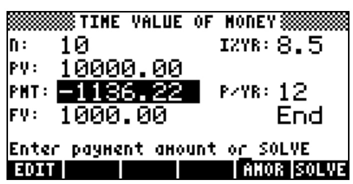
Um mit der Finanzkalkulation zu beginnen, wählen Sie mit der (▼) Taste die Position *5 Solve finance* aus. Drücken Sie die OK Taste, um die Anwendung zu starten. Es erscheint ein Eingabeformular mit Eingabefeldern für eine Anzahl von Variablen.(n, I%YR, PV, PMT, FV).



In diesem besonderen Fall können wir allen Variablen Werte zuordnen, bis auf einer, also sagen wir, n = 10, I%YR = 8.5, PV = 10000, FV = 1000, und löse für die Variable PMT (Die Bedeutung dieser Variablen wird später erklärt). Versuchen Sie folgendes:

10 OK	Eingabe von n = 10
8.5 OK	Eingabe von I%YR = 8.5
10000 OK	Eingabe von PV = 10000
▼ 1000 OK	Eingabe von FV = 1000
▲ ◄ SOLVE	Wählen und Lösen von PMT

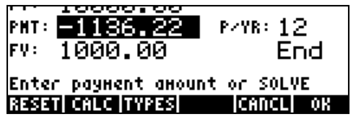
Es erscheint folgende Anzeige:



In diesem Eingabeformular werden Sie folgende Soft-Menü Tasten-Labels bemerken:

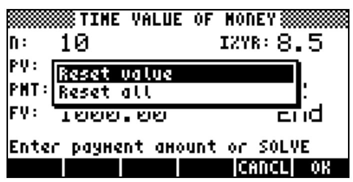
EDIT das hervorgehobene Feld editieren
AMOR Amortisations-Menü – spezielle Option für diese Anwendung
SOLVE Lösung für das hervorgehobene Feld

Durch drücken von NXT sehen wir die folgenden Soft-Menü Tasten-Labels:

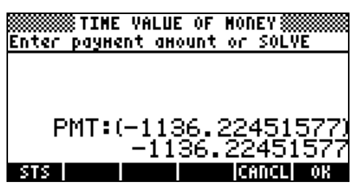


RESET Zurücksetzen aller Felder auf Standardwerte
CALC Zugriff auf den Stack für Berechnungen
TYPES Bestimmung des Objekttyps des hervorgehobenen Feldes
CANCL Operation abbrechen
OK Eingabe annehmen

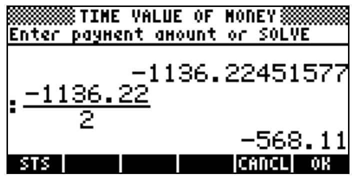
Wenn Sie die RESET Taste drücken, werden Sie aufgefordert, zwischen diesen beiden Optionen zu wählen:



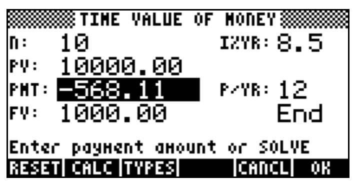
Wenn Sie *Reset value* auswählen, wird nur der hervorgehobene Wert auf den Standard-Wert zurückgesetzt Wenn Sie stattdessen *Reset all* auswählen, werden alle Felder auf ihre Standardwerte zurückgesetzt (normalerweise 0). An diesem Punkt können Sie Ihre Auswahl akzeptieren und die Aktion ausführen (durch drücken von OK), oder die Operation abbrechen (durch drücken von CANCL). Drücken Sie jetzt CANCL. Drücken CALC um auf den Stack zuzugreifen. Sie sehen folgende Anzeige:



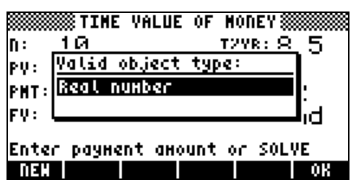
Nun haben Sie Zugriff auf den Stack und es wird der letzte hervorgehobene Wert des Eingabeformulars angezeigt Nehmen wir an, Sie möchten diesen Wert halbieren. Folgende Anzeige erscheint im ALG Modus nachdem Sie 1136.22/2 eingegeben haben:



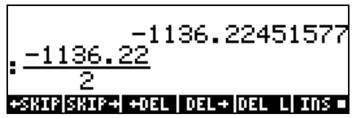
(Im RPN Modus hätten wir eingegeben: 1136.22 ENTER 2 ENTER ÷).
Drücken Sie OK um diesen neuen Wert einzugeben. Das Eingabeformular sieht nun etwa so aus:



Drücken Sie TYPES um den Datentyp im PMT Feld angezeigt zu bekommen (das hervorgehobene Feld). Sie erhalten folgende Spezifikation:



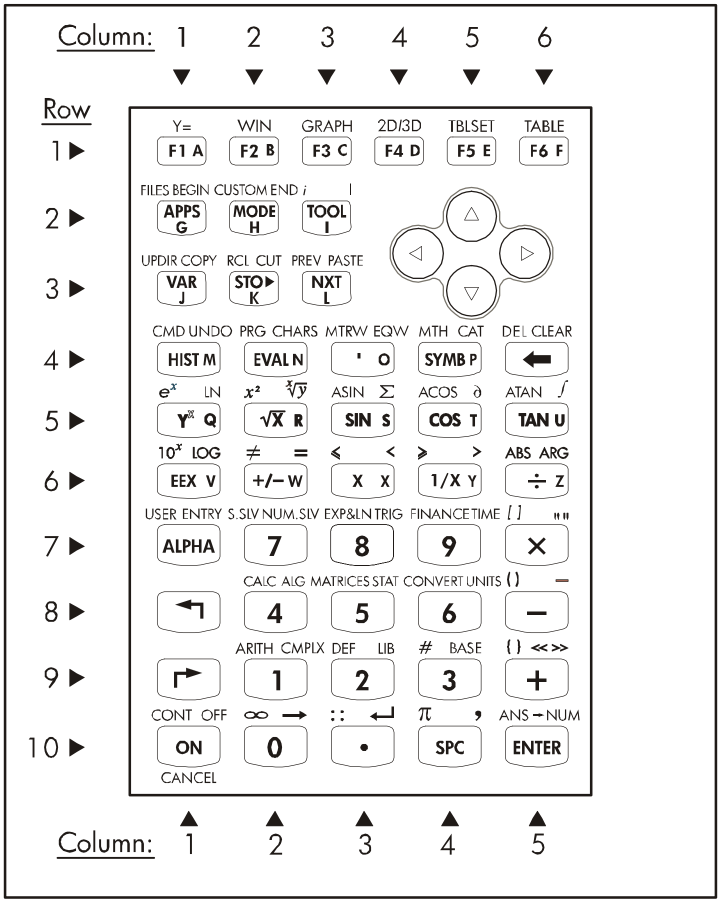
Dies zeigt an, dass der Wert des PMT Feldes eine reale Zahl sein muss. Drücken Sie OK um zum Eingabeformular zurück zu kehren und drücken Sie L um wieder ins erste Menü zu gelangen. Als nächstes drücken Sie die ENTER Taste oder die ON Taste um zum Stack zurück zu kehren. Jetzt werden die folgenden Werte angezeigt:



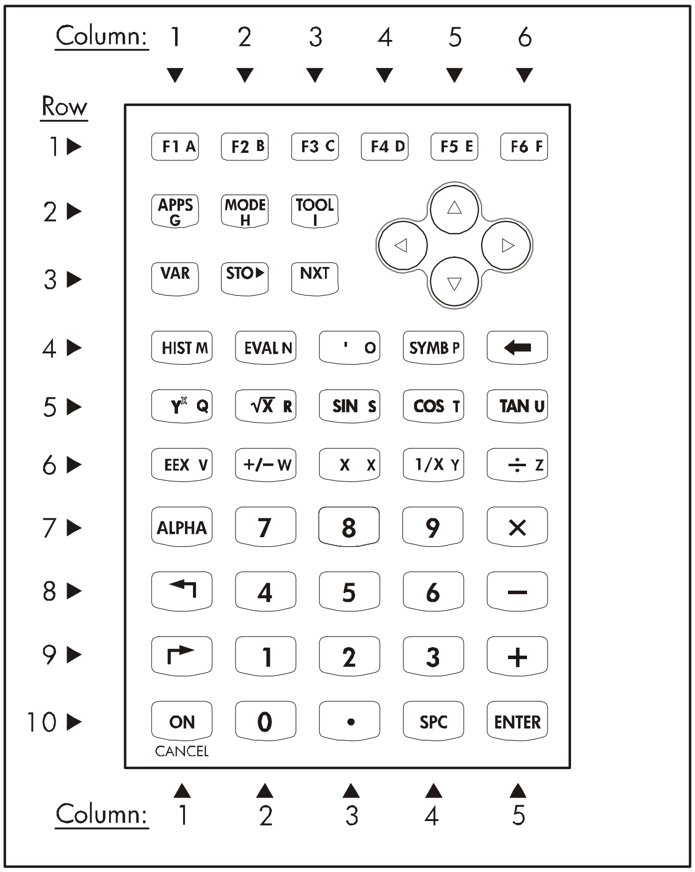
Das obere Ergebnis ist der Wert, der im ersten Teil der Übung für PMT errechnet wurde. Der zweite Wert ist das Ergebnis der Berechnung die wir durchgeführt haben, um den Wert von PMT neu zu definieren.

Anhang B
Die Tastatur des Rechners

In der nachfolgenden Abbildung sehen Sie die Tastatur Ihres Rechners mit Nummerierung der Zeilen und Spalten.



Diese Abbildung zeigt 10 Reihen von Tasten, in Kombination mit 3, 5 oder 6 Spalten.   Reihe 1 hat 6 Tasten, Reihe 2 und 3, jeweils 3, während Reihe 4 bis 10 jeweils 5 Tasten aufweisen. In der rechten oberen Ecke, in Höhe der Reihen 2 und 3 Reihe, befinden sich 4 Pfeiltasten.   Jede einzelne Taste hat drei, vier oder fünf verschiedene Funktionen. In der Abbildung unten, sehen Sie die Hauptfunktionen der Tasten. Um die Hauptfunktionen auszführen, drücken Sie einfach die entsprechende Taste.   Wir werden die Tasten je nach Reihe und Spalte in der diese sich, gemäß obiger Skizze befinden, beschreiben; somit ist *Taste (10,1)* die *ON* (Einschalt-) Taste.



Hauptfunktionen der Tasten in der Rechnertastatur

Hauptfunktionen der Tasten

Die Tasten F1 bis F6 sind mit Menüfunktionen verbunden, welche am unteren Rand des Rechner-Displays angezeigt werden. Somit können mit diesen Tasten unterschiedliche Funktionen gestartet werden, welche sind je nach aktivem Menü ändern.

- Die Pfeiltasten, ▲▼◀▶, werden dazu verwendet sich ein Zeichen in Richtung der gedrückten Taste zu bewegen (d.h. nach oben, unten, links und rechts).
- Die Funktion *APPS* startet das Anwendungsmenü.
- Die Funktion *MODE* das Menü Modus.
- *TOOL* aktiviert ein Menü verschiedener Werkzeuge zur Bearbeitung von Variablen und für die Hilfestellung.
- Die Funktion *VAR* zeigt die im aktiven Verzeichnis gespeicherten Variablen.
- *STO* wird zur Speicherung von Inhalten in Variablen verwendet.
- *NXT* wird zur Anzeige weiterer Funktionstasten oder Variablen im Verzeichnis verwendet
- Mit der Funktion HIST können Sie die History im algebraischen Modus starten, d.h. eine Sammlung aktueller Befehle in diesem Modus.
- *EVAL* wird zur Auswertung (Berechnung) algebraischer und numerischer Ausdrücke verwendet.
- Mit der Taste Apostroph ['] können Sie einen Satz von Apostrophen für algebraische Ausdrücke eingeben.
- *SYMB* aktiviert das symbolische Operationsmenü.
- Die Löschtaste, ◀, wird zum Löschen von Zeichen in einer Zeile verwendet.
- Die Taste y^x berechnet die Potenz x der Zahl y.
- $\sqrt{x}$ berechnet die Quadratwurzel einer Zahl.
- Die Tasten *SIN*, *COS* und *TAN* berechnen entsprechend den Sinus, Cosinus oder Tangens einer Zahl.
- *EEX* wird zur Eingabe der Zehnerpotenz einer Zahl verwendet (z.B. 5×10^3, wird als 5 EEX 3 eingegeben und dann als *5E3* angezeigt).
- Die Taste +/- ändert das Vorzeichen eines Eintrags.
- Mit der Taste *X* geben Sie das Zeichen X in Großbuchstaben ein.
- 1/x berechnet die Inverse einer Zahl.
- Die Tasten $+$, $-$, $\times$ und $\div$, werden zur Ausführung der Grundrechenarten eingesetzt (entsprechend Addition, Subtraktion, Multiplikation und Division).

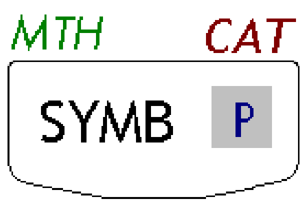
- Die *ALPHA* Taste funktioniert in Kombination mit anderen Tasten und wird zur Eingabe alphabetischer Zeichen verwendet.
- Die linke- ← bzw. die rechte → Shift-Taste wird in Kombination mit anderen Tasten zur Aktivierung von Menüs verwendet.
- Der Zahlenblock (Ziffern *0* bis *9*), wird zur Eingabe der Ziffern des Dezimalzahlen-Systems verwendet.
- Es gibt einen Dezimalpunkt (.) und eine Leertaste (*SPC*).
- Die Taste *ENTER* wird zur Eingabe einer Zahl, eines Ausdrucks oder einer Funktion zur Anzeige im Stack verwendet und
- Die Taste *ON* dient zum Einschalten des Rechners.

Funktionen der Alt-Taste

Die linke grüne *Shift-Taste*, *Taste (8,1)*, die rechte rote *Shift-Taste, Taste (9,1)* und die blaue ALPHA-*Taste, Taste (7,1)*, können mit anderen Tasten kombiniert werden, um alternative Funktionen, die auf der Tastatur angezeigt werden, zu starten. So hat z.B. die SYMB Taste, *Taste (4,4)*, die folgenden sechs Funktionen, die wie folgt zugeordnet sind:

SYMB	Hauptfunktion, startet das Menü SYMB (SYMBolic)
← MTH	Linke Shift-Funktion startet das Menü MTH (Math)
→ CAT	Rechte Shift-Funktion startet die Funktion CATalog (Katalog)
ALPHA P	ALPHA Funktion, dient zur Eingabe des Großbuchstabens P
ALPHA ← P	ALPHA-linke-Shift-Funktion, fügt den Kleinbuchstaben P ein
ALPHA → P	ALPHA-rechte-Shift-Funktion, fügt das Symbol P ein

Von den sechs dieser Taste zugeordneten Funktionen, werden nur vier davon auf der Taste selbst angezeigt. So sieht Ihre Tastenanzeige auf der Tastatur aus:



Beachten Sie Farbe und Position der Beschriftung auf der Taste, und zwar SYMB, *MTH*, *CAT* und P, zeigen die Hauptfunktion (SYMB) an und welche der drei weiteren Funktionen der jeweiligen Tastenkombination zugeordnet ist: linke-Shift Taste (*MTH*), rechte-Shift Taste (*CAT*), und ALPHA (P).

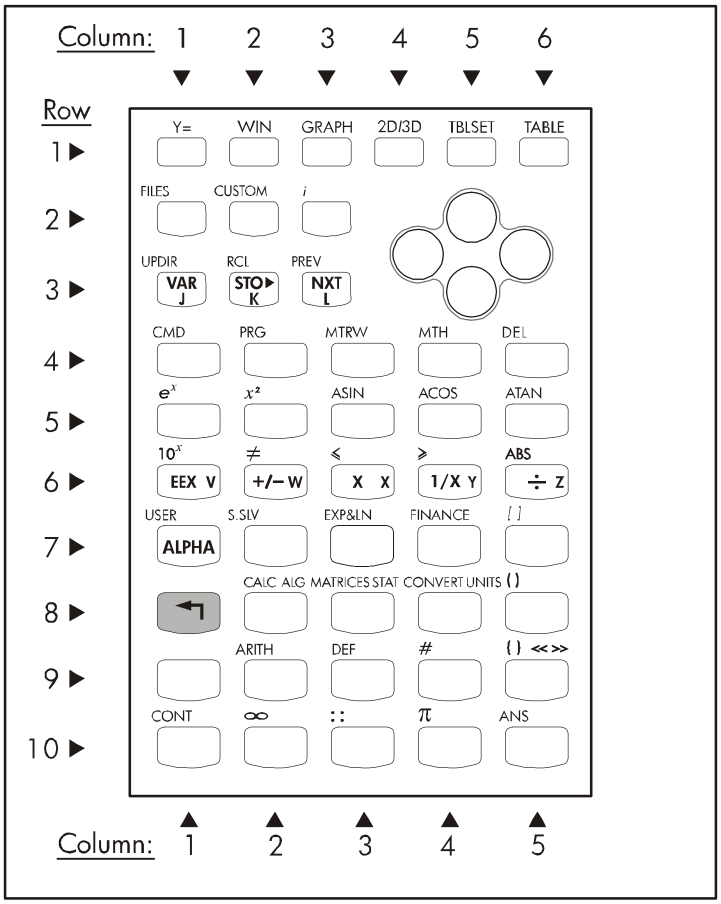
Diagramme, welche die Funktionen oder Zeichen aus Kombinationen der Rechnertasten mit der linken Shift-Taste, der rechten Shift-Taste, ALPHA, ALPHA + linke Shift-Taste und ALPHA + rechte Shift-Taste werden nachfolgend vorgestellt. In diesen Diagrammen werden die, für jede Tastenkombination resultierenden Zeichen oder Funktionen, auf weißem Hintergrund dargestellt. Ist die linke oder rechte Shift-Taste oder die Taste ALPHA aktiviert, wird diese auf schattiertem Hintergrund dargestellt. Tasten die nicht aktiviert werden können, sind auf schwarzem Hintergrund dargestellt.

Funktionen der linken Shift-Taste

In der nachfolgenden Skizze sehen Sie Funktionen, die sobald die linke- Shift-Taste, , aktiviert ist, Zeichen oder Menüs, die verschiedenen Tasten des Rechners zugeordnet sind, anzeigen.

- Die sechs Funktionen, die in Kombination mit der linken Shift-Taste und den Funktionstasten F1 bis F6 zugewiesen sind, dienen der Einstellung und Erstellung von Grafiken und Tabellen. Wenn Sie diese Funktionen innerhalb des algebraischen Rechnermodus verwenden, drücken Sie die linke Shift-Taste erst, anschließend eine der Tasten in Reihe 1. Verwenden Sie diese Funktionen im *RPN*-Modus des Rechners, müssen Sie die linke Shift-Taste gleichzeitig mit der von Ihnen ausgewählten Taste in der ersten Reihe drücken. Die Funktion *Y=* wird zur Eingabe einer Funktionen *y=f(x)* zum Plotten verwendet, die Funktion *WIN* wird zur Einstellung der Parameter für das Plot-Fenster verwendet, *GRAPH* erzeugt einen Graphen, mit der Funktion *2D/3D* wählen Sie den Typ des zu erzeugenden Graphs aus, *TBLSET* wird zur Einstellung der Parameter für eine Wertetabelle einer Funktion verwendet, mit *TABLE* können Sie eine Tabelle mit den Werten erstellen.
- *FILE* aktiviert den Datei-Browser des Rechners.
- *CUSTOM* startet die Optionen für das allgemeine Menü, das *i* wird zum Eintrag der imaginären Zahl *i* in den Stack verwendet ($i^2 = -1$).
- *UPDIR* bringt Sie in ein übergeordnetes Verzeichnis in der Struktur des Rechners.

- *RCL* wird zur Wiederherstellung von Werten von Variablen verwendet.
- *PREV* zeigt die 6 Tasten der vorhergehenden Menü-Optionen.
- Die Funktion *CMD* zeigt die neuesten Befehle, die Funktion *PRG* aktiviert die Programmiermenüs, die Funktion *MTRW* aktiviert den Matrixwriter.

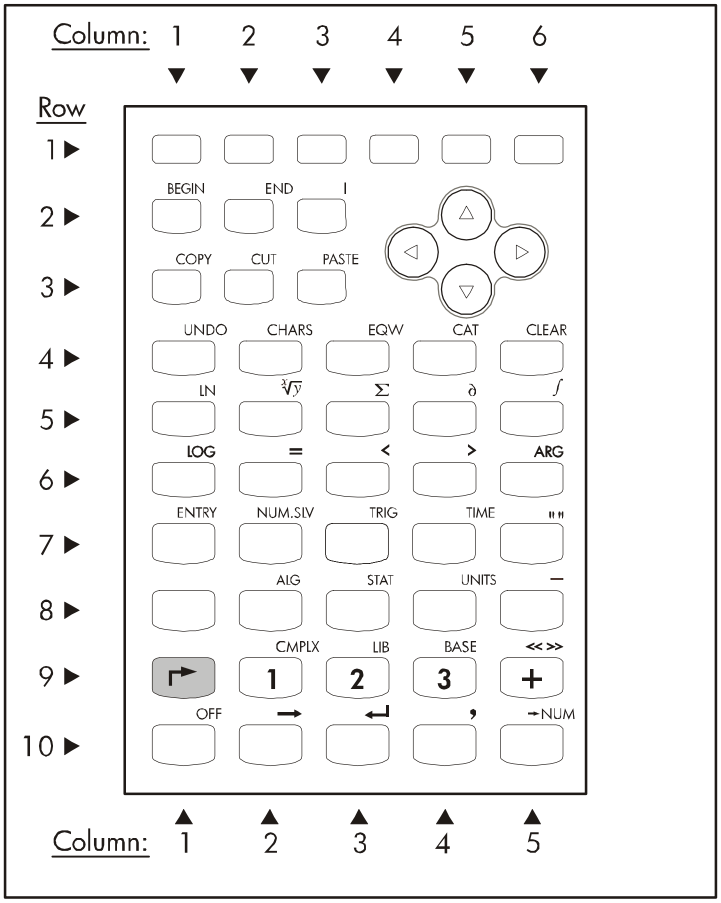


Funktionen der linken Shift-Taste ⮢ im Rechner

- *CMD* die letzten verwendeten Befehle.
- *PRG* aktiviert die Programmiermenüs.
- Das Menü MTRW startet den MatrixWriter.
- *MTH* aktiviert das Menü mit den mathematischen Funktionen.

- Die Taste *DEL* wird zum Löschen von Variablen verwendet.
- Die Taste e^x berechnet die Exponentialfunktion von x,
- x^2 berechnet das Quadrat von x (diese wird als SQ bezeichnet).
- Die Funktionen ASIN, ACOS und ATAN berechnen entsprechend die Funktionen Arcsinus, Arccosinus und Arctangens.
- Die Funktion 10^x berechnet den Antilogarithmus von x
- Die Tasten $\neq$, $\leq$ und $\geq$ werden zum Vergleich von reellen Zahlen verwendet.
- Die Funktion *ABS* berechnet den absoluten Wert einer reellen Zahl oder die Magnitude einer komplexen Zahl oder eines Vektors.
- Die Funktion *USER* aktiviert die anwenderdefinierte Menütastatur.
- Die Funktion *S.SLV* aktiviert das symbolische Löser-Menü.
- *EXP&LN* aktiviert das Menü zum Ersetzen von Gliedern in Exponential- und natürlichen Logarithmen.
- *FINANCE* aktiviert eine Menü für finanzmathematische Funktionen.
- Die Funktion *CALC* aktiviert eine Funktion von verschiedenen Rechenmethoden.
- Die *MATRICES* Funktion aktiviert eine Menü zur Erstellung und Manipulation von Matrizen.
- Die Funktion *CONVERT* aktiviert das Menü zur Konvertierung von Einheiten und anderer Ausdrücke.
- *ARITH* aktiviert ein Menü von arithmetischen Funktionen.
- Die Funktion *DEF* wird zur Definition von einfachen Funktionen als Variable im Menü des Rechners verwendet.
- Die Taste *CONT* wird zur Fortsetzung einer Rechneroperation verwendet.
- Die Taste *ANS* stellt den letzten Wert des Rechners wieder her, sofern dieser im algebraischen Modus verwendet wird.
- Die Tasten *[ ]*, *( )* und *{ }* werden zum Einfügen von Klammern verwendet, oder ….
- Die # Taste zur Eingabe von Nummern außerhalb der aktiven Zahlenbasis.
- Das Zeichen ∞ wird zur Eingabe des Zeichens Unendlich in einen Ausdruck verwendet.
- Die Taste Pi π wird zur Eingabe des Symbols π verwendet (das Verhältnis der Länge eines Kreisumfangs zu seinem Radius).

- Werden die Pfeiltasten mit der linken Shift-Taste kombiniert, wird der Cursor zum ersten Zeichen in Richtung der gedrückten Taste bewegt.



Funktionen der rechten Shift-Taste ⤷ im Rechner

Funktionen der rechten Shift-Taste

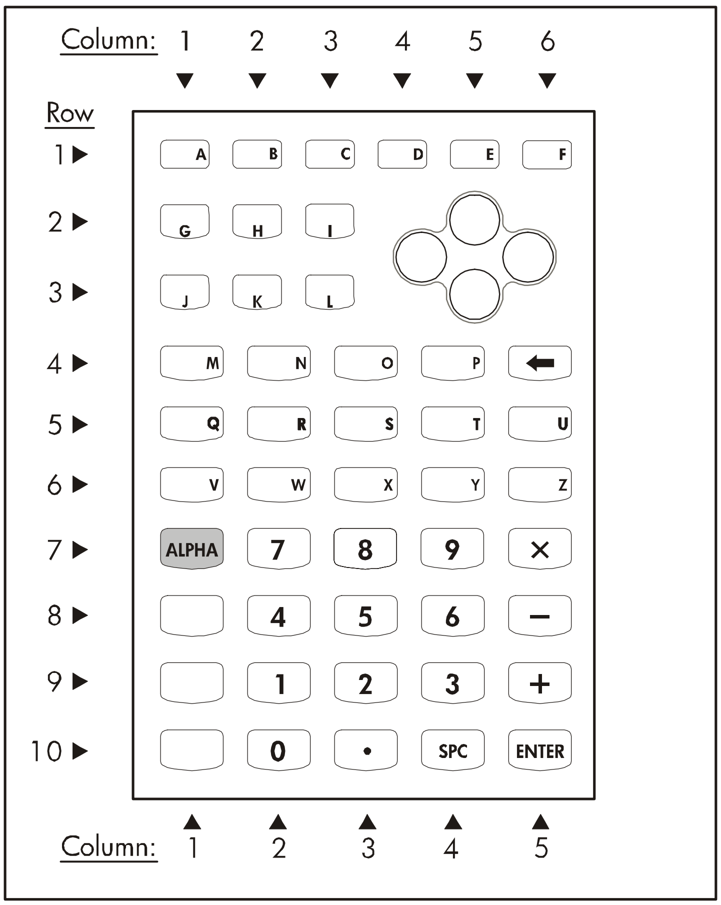
In der obigen Skizze sehen Sie Funktionen, Zeichen oder Menüs, die verschiedenen Tasten des Rechners zugeordnet sind, sobald die rechte Shift-Taste, ⤷, aktiviert ist.

- Die Funktionen *BEGIN, END, COPY, CUT* und *PASTE* werden zur Bearbeitung verwendet.

- Die Taste *UNDO* wird zur Wiederherstellung der letzten Operation im Rechner verwendet.
- Die Funktion *CHARS* aktiviert das Menü für Sonderzeichen.
- Die Funktion *EQW* dient zum Starten des EquationWriters.
- Die Funktion *CAT* wird zum Aufrufen des Katalogs verwendet.
- Die Funktion *CLEAR* bereinigt die Tastatur.
- *LN* berechnet den natürlichen Logarithmus.
- Die Funktion $\sqrt[x]{y}$ berechnet die x-te Wurzel von y.
- Die Funktion Σ dient zur Summenberechnung (oder der große griechische Buchstabe Sigma).
- Die Funktion ∂ berechnet die Ableitung.
- Die Funktion ∫ berechnet die Integrale.
- *LOG* berechnet den Zehnerlogarithmus.
- *ARG* berechnet das Argument einer komplexen Zahl.
- Die Funktion *ENTRY* ändert den Eingabemodus in der Bearbeitung.
- *NUM.SLV* startet den numerischen Löser.
- *TRIG* startet die trigonometrischen Austauschfunktionen.
- Die Funktion *TIME* startet das Zeitmenü.
- *ALG* startet das algebraische Menü. *STAT* startet das statistische Menü.
- Die Funktion *UNITS* startet das Menü für Maßeinheiten.
- Die *CMPLX* Funktion startet das Menü für komplexe Zahlen.
- *LIB* aktiviert die Bibliotheksfunktionen.
- *BASE* aktiviert das numerische Menü zur Konvertierung von Basen.
- Die Taste OFF schaltet den Rechner aus.
- Die Taste *→NUM* erzeugt einen numerischen (oder Gleitpunkt) Wert eines Ausdrucks.
- Mit den Tasten " " können Sie einen paar Anführungszeichen eingeben, wird bei der Eingabe von Alpha-Zeichenketten verwendet.
- Die Taste __ fügt einen Unterstrich ein.
- Die << >> Taste gibt das Symbol für ein Programm ein.
- Die Taste → fügt einen Pfeil ein, stellt eine Programmeingabe dar.
- Die Taste ↵ erzeugt eine Zeilenschaltung in Programmen oder Alpha-Zeichenketten.
- Die Taste Komma (,) fügt ein Komma ein.
- Werden die Pfeiltasten mit der rechten Shift-Taste kombiniert, wird der Cursor zu dem am weitesten entfernten Zeichen in Richtung der gedrückten Taste bewegt.

ALPHA Zeichen

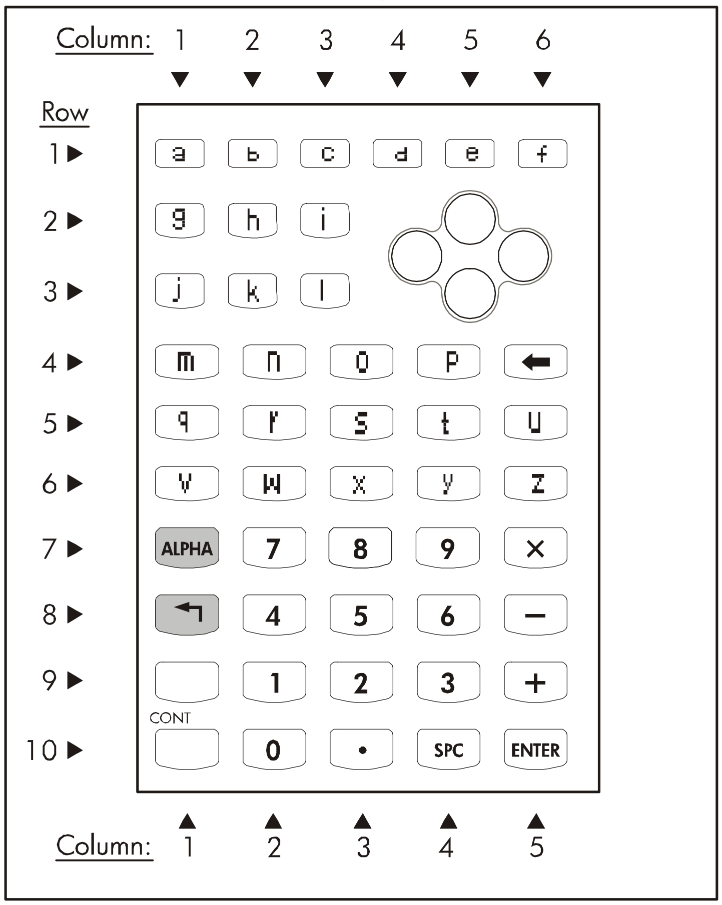
Nachfolgende Skizze zeigt die Zeichen, welche verschiedenen Rechnertasten zugeordnet sind, wenn ALPHA ALPHA aktiviert ist. Beachten Sie, dass die ALPHA Funktion hauptsächlich zur Eingabe von Großbuchstaben des englischen Alphabets (von *A* bis *Z*) verwendet werden. Die Zahlen, mathematischen Symbole (-, +), Dezimalpunkt (.) und die Leertaste (*SPC*) haben die gleichen Funktionen wie die Hauptfunktionen dieser Tasten. Die ALPHA Funktion erzeugt ein Sternchen (*), wenn diese mit dem Malzeichen kombiniert wird, d.h. ALPHA ×.



Alpha ALPHA Funktionen der Rechnertastatur

Zeichen mit Alpha und linke Shift-Taste

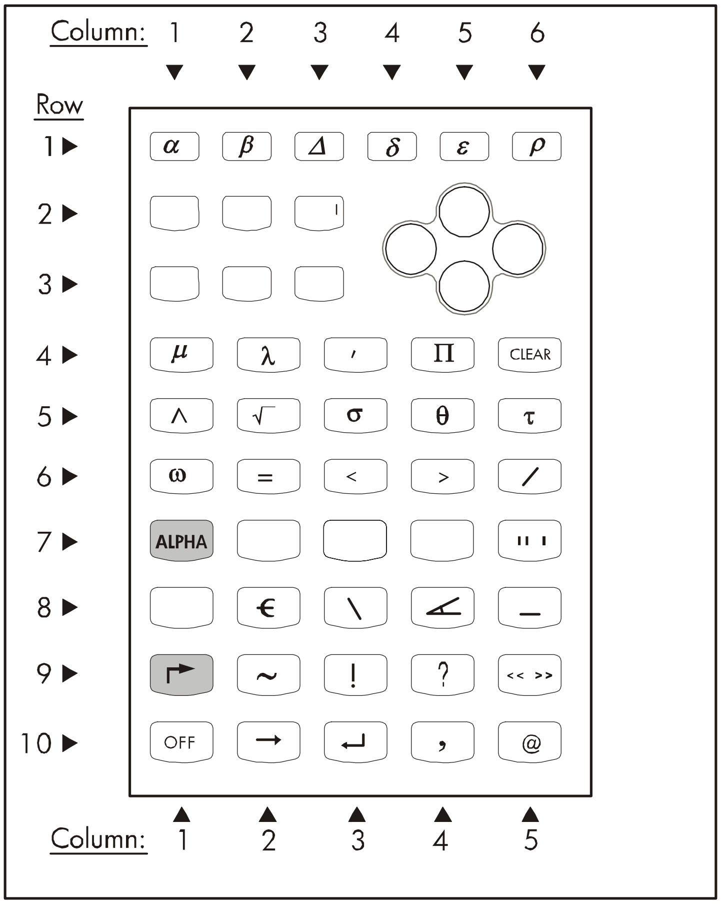
Nachfolgende Skizze zeigt die Zeichen, welche verschiedenen Rechnertasten zugeordnet sind, wenn ALPHA [ALPHA] mit der linken Shift-Taste [←] kombiniert wird. Beachten Sie, dass die [ALPHA][←] Kombination hauptsächlich zur Eingabe von Kleinbuchstaben des englischen Alphabets (von *A* bis *Z*) verwendet wird. Die Zahlen, mathematischen Symbole (-, *+*, *×*), Dezimalpunkt (.) und die Leertaste (*SPC*) haben die gleichen Funktionen wie die Hauptfunktionen dieser Tasten. Die Tasten ENTER und CONT funktionieren wie deren Hauptfunktion, auch dann wenn die Kombination [ALPHA][←] verwendet wird.



Alpha [ALPHA][←] Funktionen der Rechnertastatur

Zeichen mit Alpha und rechte Shift-Taste

Nachfolgende Skizze zeigt die Zeichen, welche verschiedenen Rechnertasten zugeordnet sind, wenn ALPHA ALPHA mit der rechten Shift-Taste ↱ kombiniert wird.



Alpha ALPHA ↱ Funktionen der Rechnertastatur

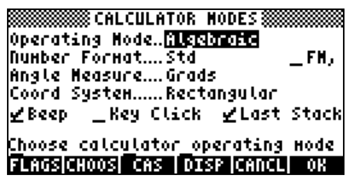
Beachten Sie, dass die Kombination ALPHA ↱ hauptsächlich zur Eingabe von Sonderzeichen in den Stack verwendet wird. Die Tasten CLEAR, OFF, →, ↵, Komma (,), sowie die OFF Taste arbeiten in deren ursprünglicher Funktion, auch dann wenn die Kombination ALPHA ↱ verwendet wird. Die

Sonderzeichen, die Sie mit der Kombination ALPHA ↱ eingeben können beinhalten griechische Buchstaben (α, β, Δ, δ, ε, ρ, μ, λ, σ, θ, τ, ω, und Π); weitere Sonderzeichen die Sie mit der Kombination ALPHA ↱ eingeben können sind |, ', ^, =, <, >, /, ", \, __, ~, !, ?, <<>> und @.

Anhang C
CAS Einstellungen

CAS steht für Computer Algebraisches System. Dies ist das mathematische Herzstück des Rechners, in welchem die symbolischen mathematischen Operationen und Funktionen programmiert sind. Das CAS bietet eine Reihe von Einstellungen, die nach Typ oder Operationsart eingestellt werden können. Um die möglichen CAS Einstellungen anzusehen, gehen Sie wie folgt vor:

- Drücken Sie die Schaltfläche MODE um die CALCULATOR MODES Eingabemaske zu starten.



Am unteren Teil des Displays finden Sie nachfolgende Funktionstasten:
Optionen:

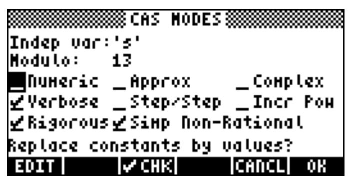
FLAGS	stellt ein Menu zur Manipulation von Flags (*) im Rechner zur Verfügung
CHOOS	ermöglicht es dem Anwender die verschiedenen Felder im Formular auszuwählen
CAS	stellt eine Eingabemaske zur Änderung der CAS Einstellungen berei
DISP	stellt eine Eingabemaske zur Änderung der Display Einstellungen bereit
CANCL	Schließt die aktuelle Eingabemaske und wechselt zur Normalanzeige
OK	verwenden Sie diese Taste zum Bestätigen Ihrer Einstellungen

(*) Flags sind Variablen im Rechner, welche über Zahlen angesprochen, entweder "gesetzt" oder "nicht gesetzt" sein können, bestimmte Anwendungsoptionen des Rechners ändern.

Drücken Sie die Taste NXT erhalten Sie eine Anzeige der noch verbleibenden Optionen in der CALCULATOR MODES Eingabemaske:

RESET der Anwender kann eine hervorgehobene Option rückgängig machen
CANCL Schließt die aktuelle Eingabemaske und wechselt zur Normalanzeige
OK verwenden Sie diese Taste zum Bestätigen Ihrer Einstellungen

- Um zum ursprünglichen Menü der CALCULATOR MODES Eingabemaske zurückzukehren, drücken Sie die Taste NXT. Interessant für uns an dieser Stelle sind die Änderungen der CAS-Einstellungen. Die wird durch drücken der Funktionstaste CAS erreicht. Die voreingestellten Werte der CAS-Einstellungen werden nachfolgend aufgezeigt:



- Benutzen Sie die Pfeiltasten, um zwischen den einzelnen Optionen der CAS MODES Eingabemaske zu navigieren: ◀ ▶ ▼ ▲.

- Um irgendeine der obigen Einstellungen aus- oder abzuwählen, wählen Sie zuerst den Unterstrich vor der von Ihnen gewünschten Option und bewegen Sie die CHK Funktionstaste solange, bis die gewünschte Einstellung erreicht ist. Sobald eine Option gewählt wurde, erscheint ein Häkchen über dem Unterstrich (d.h. im obigen Beispiel die Option *Rigorous* und *Simp Non-Rational*). Nicht ausgewählte Optionen enthalten kein Häkchen über dem Unterstrich vor der Option (im obigen Beispiel sind das die Optionen *_Numeric, _Approx, _Complex, _Verbose, _Step/Step, _Incr Pow*).

- Nachdem Sie nun die gewünschten Optionen, welche Sie für die Eingabemaske des CAS MODES haben möchten, ausgewählt haben, drücken Sie die Funktionstaste OK. Somit kehren Sie zur Eingabemaske des CALCULATOR MODES zurück. Um zur Normalansicht des Rechners zurückzukehren, drücken Sie die Taste OK ein weiteres Mal.

Auswahl der unabhängigen Variablen

Viele der vom CAS zur Verfügung gestellten Funktionen verwenden eine vordefinierte unabhängige Variable. Standardmäßig wird eine solche Variable mit dem Großbuchstaben *X*, wie in der nachfolgenden CAS MODES Eingabemaske angezeigt, ausgewählt. Der Anwender kann diesen Buchstaben, aber mit jedem anderen Buchstaben oder Kombination von Zahlen und Buchstaben (eine Variable muss mit einem Buchstaben beginnen) ersetzen, indem er das Feld *Indep var* (unabhängige Variable) in der CAS MODES Eingabemaske ändert.

Im Verzeichnis {HOME CASDIR} existiert eine VX genannte Variable die standardmäßig den Wert 'X' annimmt. Dies ist der bevorzugte Name für die unabhängige Variable in algebraischen und Calculus Anwendungen. Deshalb wird für die meisten Übungen dieses Kapitels, X als die unbekannte Variable verwendet. Benutzen Sie eine andere unabhängige Variable, z.B. mit der HORNERschen Funktion, wird das CAS nicht richtig funktionieren.

Die Variable VX hat ihren festen Platz im Verzeichnis {HOME CASDIR}. Weitere CAS Variablen in Ihrem {HOME CASDIR} sind z.B. REALASSUME (REALA), MODULO (MODUL), CASINFO (CASIN), usw.

Sie können den Wert der Variablen VX ändern, indem Sie einen neuen algebraischen Namen in dieser speichern, z.B. ‘x’, ‘y’, ‘m’, usw. Vorzugsweise sollten Sie aber 'X' als Ihre Variable VX für die Beispiele in diesem Handbuch beibehalten.

Vermeiden Sie in Ihren Programmen oder Gleichungen die Variable VX zu verwenden, um diese nicht mit der CAS Variablen VX zu verwechseln. Wenn Sie sich aber auf die x-Komponente der Geschwindigkeit beziehen wollen, können Sie dafür entweder vx oder Vx benutzen.

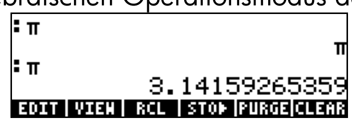
Modul auswählen

Die Option *Modulo* in der CAS MODES Eingabebox, stellt eine Zahl (standardmäßig = 13) dar, die in der modularen Arithmetik verwendet wird. Weitere Details zur modularen Arithmetik werden an anderer Stelle beschrieben.

Numerischer vs. symbolischer CAS-Modus

Wird der *Numeric* (numerische) CAS Modus ausgewählt, werden bestimmte vordefinierte Konstanten des Rechners mit deren komplettem Gleitpunktwert angezeigt. Die Option _*Numeric* ist standarmäßig nicht ausgewählt, was bedeutet, dass die vordefinierten Konstanten als Symbol, anstelle deren Werte, im Display angezeigt werden.

Die nachfolgende Abbildung zeigt die Werte der Konstanten π (dem Verhältnis zwischen Länge des Kreisumfangs zum Radius) in symbolischem Format, gefolgt von deren numerischem oder Gleitpunkt-Format. Dieses Beispiel entspricht dem algebraischen Operationsmodus des Rechners.



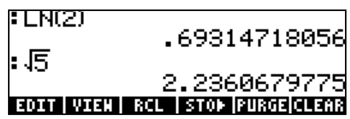
Das gleiche Beispiel im RPN-Modus wird als nächstes angezeigt:

Näherungs- vs. exakter CAS-Modus

Ist der _*Approx* (Näherungs) Modus ausgewählt, werden symbolische Operationen (z.B. bestimmte Integrale, Quadratwurzeln usw.) numerisch berechnet. Ist der _*Approx* Modus nicht ausgewählt (der Exakt-Modus ist aktiv), werden symbolische Operationen als geschlossene algebraische Ausdrücke, wann immer dies möglich ist, berechnet.

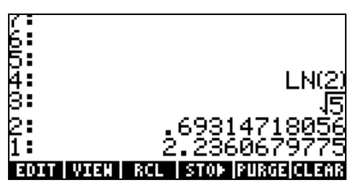
Nachfolgende Abbildung zeigt eine Reihe von symbolischen Ausdrücken, welche mit aktiviertem algebraischen Modus eingegeben wurden:

Im algebraischen Modus, werden die vom Anwender eingegebenen Objekte auf der linken Seite des Displays angezeigt, gefolgt vom Ergebnis auf der rechten Seite der Anzeige. Die oben gezeigten Ergebnisse zeigen die symbolischen Ausdrücke für *ln(2)*, d.h. der natürliche Logarithmus von 2, und $\sqrt{5}$, d.h. die Quadratwurzel von 5. Ist die *_Numeric* (numerische) Option im CAS ausgewählt, werden die nachfolgenden Operationen wie folgt angezeigt:



Die zur Eingabe dieser Werte nötigen Tastenanschläge im algebraischen Modus sind folgende: [↱] LN [2] [ENTER] [√x] [5] [ENTER]

Die gleichen Berechnungen können auch im RPN Modus erstellt werden. Stack Ebene *3:* und *4:* zeigen in diesem Fall die exakten CAS Einstellungen (d.h. die *_Numeric* CAS Option ist nicht ausgewählt), während in Stack Ebene *1:* und *2:* den Fall veranschaulichen, wenn die *Numeric* CAS Option ausgewählt ist.



Die dazu notwendigen Tastenanschläge sind: [2] [↱] LN [5] [√x]

Die Abkürzung auf der Tastatur zum schnellen Wechsel zwischen dem APPROX und EXACT Modus ist, wenn Sie die rechte Shift-Taste drücken, diese halten und gleichzeitig die Taste ENTER drücken, d.h.
↱ (halten) ENTER.

Reelle Zahlen vs. Integerzahlen

In CAS Operationen werden Integerzahlen verwendet, um die volle Genauigkeit bei Berechnungen beizubehalten. Reelle Zahlen werden in Form einer Mantisse und einem Exponenten gespeichert und haben eine begrenzte Präzision. Im APPROX Modus hingegen, wann immer Sie eine Integerzahl eingeben, wird diese automatisch in deren reelle Zahl geändert, wie Sie nachfolgend sehen können:

Wann immer der Rechner eine Integerzahl gefolgt von einem Dezimalpunkt anzeigt, bedeutet dies, dass der Rechner diese in ihre reelle Darstellung umgewandelt hat. Dies bedeutet, der APPROX Modus im Rechner war bei der Eingabe der Zahl eingestellt.

Es ist ratsam den EXACT Modus als Standard CAS-Modus auszuwählen und dann in den APPROX Modus zu wechseln, wenn der Rechner dies bei der Durchführung einer Operation benötigt.

Zusätzliche Informationen zu reellen und Intergerzahlen, wie auch anderen Objekten im Rechner, finden Sie in Kapitel 2.

Komplex vs. reeller CAS-Modus

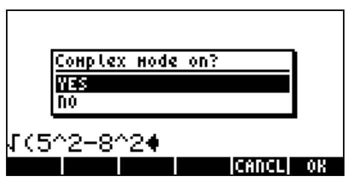
Eine komplexe Zahl ist eine Zahl *a+bi*, wobei *i*, definiert durch $i^2 = -1$ die imaginäre Einheit darstellt (Elektroingenieure ziehen das *j* dafür vor) und *a* und *b* reelle Zahlen sind. So z.B. ist die Zahl *2 + 3i* eine komplexe Zahl.

Zusätzliche Informationen zu komplexen Zahlen finden Sie in Kapitel 4 dieses Handbuches.

Ist die CAS-Option _*Complex* ausgewählt und das Ergebnis der Berechnung eine komplexe Zahl, dann wird das Ergebnis entweder als *a+bi* oder als geordnetes Paar (*a,b*) angezeigt. Andererseits, ist der _*Complex* Modus nicht ausgewählt, (d.h. der REAL Modus ist aktiv) und das Ergebnis der Berechnung eine komplexe Zahl, werden Sie gebeten in den Complex Modus zu wechseln. Bestätigen Sie den Wechsel nicht, erhalten Sie eine Fehlermeldung.

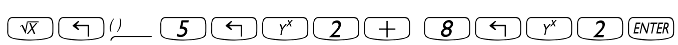
Beachten Sie, dass im COMPLEX Modus das CAS breiter gefächerte Operationen als im REAL Modus durchführen kann, aber auch dementsprechend langsamer sein wird. Es ist ratsam den REAL Modus als Standard CAS-Modus auszuwählen und dann in den COMPLEX Modus zu wechseln, wenn der Rechner dies bei der Durchführung einer Operation benötigt.

Das nachfolgende Beispiel zeigt die Berechnung der Menge $\sqrt{5^2 - 8^2}$ im algebraischen Modus, mit ausgewählter Real Option des CAS. In diesem Fall, werden Sie gefragt, ob Sie in den Complex Modus wechseln wollen:

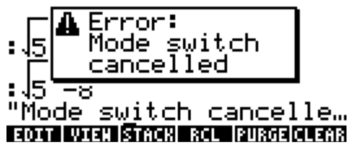


Drücken Sie nun die Funktionstaste () (OK) drücken, dann wird die _Complex Option erzwungen und das Ergebnis sieht wie folgt aus:

Die Tastenanschläge, die oben verwendet wurden sind:



Wenn Sie gebeten werden in den COMPLEX Modus zu wechseln, benutzen Sie die Taste F6 . Wollen Sie den Wechsel in den COMPLEX-Modus nicht durchführen, erhalten Sie nachfolgende Fehlermeldung:



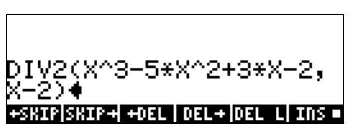
Ausführlicher vs. kurzer CAS Modus

Ist die *_Verbose* (ausführliche) CAS-Option ausgewählt, werden verschiedene Anwendungen mit Kommentaren im Display ausgegeben. Ist diese Option nicht ausgewählt, erhalten Sie keine Kommentarzeilen zu jenen Anwendungen. Die Kommentarzeilen erscheinen, während der Durchführung der Rechenoperation, in den oberen Zeilen des Rechners.

Step-by-step CAS Modus

Ist die *_Step/step* (schrittweise) Option ausgewählt, werden bestimmte Operationen im Einzelschrittmodus auf dem Display angezeigt. Ist die *_Step/step* Option nicht ausgewählt, werden keine Zwischenberechnungen angezeigt.

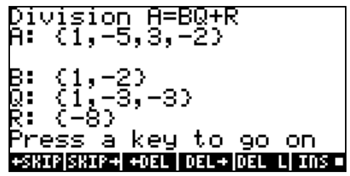
Bei ausgewählter Step/step Option z.B. werden die nachfolgenden Anzeigen eine schrittweise Division zweier Polynome, und zwar $(X^3\text{-}5X^2\text{+}3X\text{-}2)/(X\text{-}2)$ anzeigen. Dies erreicht man durch Anwendung der Funktion DIV2, wie nachfolgend gezeigt. Drücken Sie ENTER, um den ersten Schritt anzuzeigen:

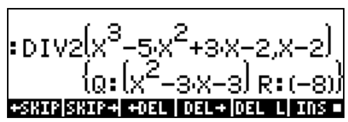


Die Anzeige verrät uns, dass der Rechner eine Division zweier Polynome A/B durchführt, so dass A = BQ + R, wobei Q den Quotienten und R den Restwert darstellt. Für den vorliegenden Fall, $A = X^3\text{-}5X^2\text{+}3X\text{-}2$ und $B = X\text{-}2$. Diese

Polynome werden im Display als Auflistung ihrer Koeffizienten dargestellt. So z.B. stellt der Ausdruck A: {1,-5,3,-2} das Polynom A = X^3-5X^2+3X-2, B:{1,-2} das Polynom B = X-2, Q: {1} das Polynom Q = X und R:{-3,3,-2} das Polynom R = -3X^2+3X-2 dar.

Drücken Sie an dieser Stelle z.B. die Taste ENTER. Drücken Sie ENTER weiter, um die zusätzlichen Schritte zu erzeugen:





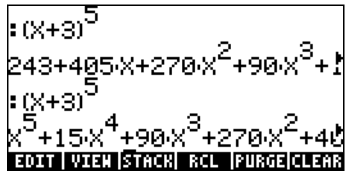
Somit stellen die Zwischenschritte Koeffizienten der Quotienten und Restwerte der schrittweisen synthetischen Division dar, so als wären diese manuell durchgeführt worden, d.h.

$$\frac{X^3-5X^2+3X-2}{X-2} = X^2 + \frac{-3X^2+3X-2}{X-2} =$$

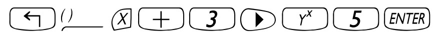
$$X^2-3X+\frac{-3X-2}{X-2} = X^2-3X-3X-\frac{8}{X-2}.$$

Erhöhen der Potenzen im CAS-Modus

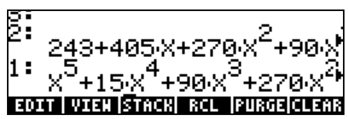
Ist die CAS Option *_Incr pow* ausgewählt, werden die Polynome so angezeigt, dass die Glieder in aufsteigender Potenz der unabhängigen Variablen dargestellt werden. Ist die CAS Option *_Incr pow* nicht ausgewählt, werden die Polynome so angezeigt, dass die Glieder in absteigender Potenz der unabhängigen Variablen dargestellt werden. Nachfolgend ein Beispiel im algebraischen Modus:



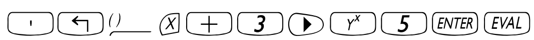
Im ersten Fall, wird das Polynom $(X+3)^5$ in aufsteigender Folge der Potenz von X aufgelistet, während im zweiten Beispiel das Polynom mit absteigender Reihenfolge der Potenz von X angezeigt wird. Die für beide Fälle notwendigen Tastenanschläge sind:



Im ersten Fall war die Option *_Incr pow* ausgewählt, während diese im zweiten Fall nicht ausgewählt war. Das gleiche Beispiel im RPN-Modus, nachfolgend:



Die gleiche Tastenfolge wurde für dieses Ergebnis verwendet:



Genaue CAS Einstellung

Ist Ihre CAS Einstellung auf *_Rigorous* eingestellt, wird der algebraische Ausdruck *|X|* d.h. der absolute Wert nicht auf *X* vereinfacht. Ist die Option *_Rigorous* nicht ausgewählt, wird der algebraische Ausdruck *|X|* auf *X* vereinfacht.

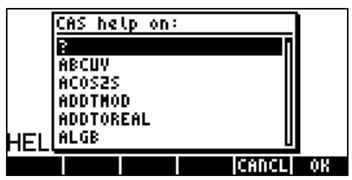
Das CAS kann eine größere Vielzahl von Problemen lösen, wenn der genaue Modus nicht eingestellt ist. Das Ergebnis hingegen oder die Domäne, in welcher die Ergebnisse angewendet werden können, könnte eingeschränkter sein.

Vereinfachung nicht rationellen CAS-Einstellungen

Ist die Option *_Simp Non-Rational* ausgewählt, werden nicht-rationelle Ausdrücke automatisch vereinfacht. Ist die Option *_Simp Non-Rational* nicht ausgewählt, werden nicht-rationelle Ausdrücke nicht automatisch vereinfacht.

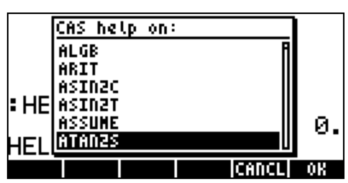
Verwenden der CAS Hilfe-Funktion

Schalten Sie den Rechner ein und drücken Sie die Taste TOOL, um das TOOL Menü zu starten. Als nächstes drücken Sie die Funktionstaste F2 gefolgt von der Taste ENTER (die Taste ganz unten rechts auf der Tastatur des Rechners), um die Hilfefunktion (HELP) zu aktivieren. Die Anzeige sieht wie folgt aus:



An dieser Stelle erhalten Sie eine Liste aller CAS Befehle in alphabetischer Reihenfolge: Verwenden Sie die Pfeiltaste ▼ zur Navigation durch die Liste. Um in der Liste nach oben zu gehen, verwenden Sie die Pfeiltaste ▲. Die Pfeiltasten befinden sich auf der rechten Seite des Display zwischen der ersten und vierten Zeile in Ihrem Rechner.

Angenommen Sie suchen Informationen zum Befehl ATAN2S (Arctangens-zu-Sinus Funktion). Drücken Sie die Pfeiltaste ▼, so lange, bis der Befehl ATAN2S in der Liste hervorgehoben ist:



Beachten Sie, dass in dieser Instanz nur den Funktionstasten F5 und F6 Befehle zugeordnet sind, und zwar:

CANCL F5 CANCeL (abbrechen) der Hilfefunktion
OK F6 OK zur Aktivierung der Hilfefunktion für den ausgewählten Befehl

Drücken Sie die Taste CANCL F5 wird die HELP (Hilfe) Funktion übergangen und der Rechner kehrt zur Normalanzeige zurück.

Um die Auswirkung der OK Taste in der HELP Funktion zu sehen, wiederholen wir die oben verwendeten Schritte zur Auswahl des Befehls ATAN2S aus der Liste der CAS Befehle: HELP F2 ENTER ▼ ▼ …(10 mal)

Um Informationen zum Befehl ATAN2S zu erhalten, drücken Sie die Taste OK F6 .

Die Hilfefunktion zeigt an, dass der Befehl oder die Funktion ATAN2S den Wert von *atan(x)*, der Arctangenswert eines Wertes *x*, durch dessen Äquivalente als *asin* (Arcsinus) Funktion darstellt, d.h.

die vierte und fünfte Zeile im Display zeigt ein Anwendungsbeispiel der Funktion ATAN2S. Zeile vier, und zwar ATAN2S(ATAN(X)) ist die Aussage der Operation die durchgeführt werden soll, während Zeile fünf, und zwar ASIN(X/√(X^2+1)) das Ergebnis darstellt.

Die untere Zeile im Display, welche mit dem Wort *See*: (siehe) beginnt, ist eine Referenz zu einem Listing weiterer verwandter CAS Befehle zu ATAN2S im Rechner.

Beachten Sie, dass in diesem Fall sechs Befehle den Funktionstasten zugeordnet sind (Sie können dies mit der Taste NXT überprüfen und werden feststellen, dass keine weiteren Menüeinträge vorhanden sind). Die Befehle der Funktionstasten sind wie folgt:

EXIT F1 EXIT (verlassen) der Hilfefunktion
ECHO F2 kopieren des Beispielbefehls in den Stack und verlassen
SEE1 F3 Siehe den ersten Link (falls überhaupt einer existiert) in der Referenzliste.

SEE2 F4 Siehe den zweiten Link (falls überhaupt einer existiert) in der Referenzliste.

SEE3 F5 Siehe den dritten Link (falls überhaupt einer existiert) in der Referenzliste

MAIN F6 Rückkehr zur MAIN (Haupt) Befehleliste der Hilfefunktion

In diesem Fall wollen wir das Beispiel in den Stack ECHOen (kopieren), durch drücken der Tasten ECHO F2. Die daraus resultierende Anzeige sieht wie folgt aus.

Nun zeigen vier Zeilen des Displays die Ausgabe an. Die ersten beiden Zeilen von oben entsprechen dem ersten Beispiel aus der HELP Funktion, in welcher wir die Hilfeanforderung abbrechen. Die dritte Zeile von oben zeigt den aktuellen Aufruf der HELP Funktion, während die letzte Zeile das ECHO des Beispielbefehls anzeigt. Um das Kommando zu aktivieren drücken Sie die Taste ENTER. Die Lösung lautet:

Beachten Sie, dass sobald neue Zeilen erzeugt werden, das Display (oder der Stack), die bestehenden Zeilen nach oben verschiebt und den unteren Teil des Displays mit mehr Ausgabezeilen füllt.

Die in diesem Abschnitt beschriebene HELP Funktion, ist bei der Definition von vielen CAS Befehlen im Rechner vorhanden. Jeder Eintrag in der CAS Hilfefunktion, wann immer angemessen, beinhaltet auch ein Beispiel des Befehls, sowie Referenzen, wie in diesem Beispiel gezeigt.

Um direkt einen bestimmten Befehl in der Hilfefunktion anzusteuern, ohne die Pfeiltasten dafür zu benutzen, kann auch nur der erste Buchstaben dieses Befehls eingegeben werden. Angenommen, Sie möchten Informationen zum Befehl IBP (Integration by Parts – Integration durch Teile) erhalten, können Sie als Erstes die ALPHA Taste, sobald die Funktion Hilfe gestartet ist, (erste Taste in der vierten Reihe von unten) gefolgt von der Taste i (die gleiche Taste wie die TOOL Taste, d.h. ALPHA I) drücken. Dies bringt Sie automatisch auf den ersten Befehl der mit dem Buchstaben *i* beginnt, und zwar IBASIS. Wenn Sie dann die Pfeiltaste ▼ zweimal drücken, befinden Sie sich auf dem IBP Befehl. Durch drücken der Tasten OK F6 wird die Hilfefunktion für diesen Befehl gestartet. Das Drücken der Tasten CANCL F6 bringt Sie zur Befehlsübersicht zurück oder EXIT F1 , um die Hilfefunktion zu verlassen.

Referenzen für nicht-CAS Befehle

Die Hilfefunktion enthält Einträge für alle für das CAS (Computer Algebraic System – algebraisches System des Rechners) entwickelten Befehle. Es gibt eine große Anzahl anderer Funktionen und Befehle, welche ursprünglich für die hp 48g Reihe entwickelt wurden, welche aber nicht in der Hilfefunktion enthalten sind. Gute Referenzen für jene Befehle finden Sie im Referenzhandbuch der hp 48g Reihe (HP Teile Nr. 00048-90126) und dem fortgeschrittenen Anwenderhandbuch (HP Teile Nr. 00048-90136), beide 1993 von Hewlett-Packard Company, Corvallis, Oregon veröffentlicht.

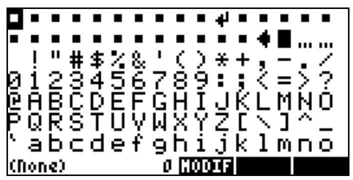
Nutzungsbedingungen für CAS

Die Anwendung von CAS setzt beim Benutzer ein fundiertes mathematisches Wissen voraus. Die CAS-Software wird ohne gesetzliche Garantie geliefert. Wenn nicht anders schriftlich erwähnt, wird die Software vom Copyright-Besitzer ohne Mängelgewähr und ohne Garantie, weder stillschweigender noch zugesagter Art, zur Verfügung gestellt, einschließlich aber nicht beschränkt auf Garantien über Gebrauchstauglichkeit und Eignung für einen bestimmten Zweck. Qualität und Leistung betreffende Risiken der CAS-Software liegen beim Anwender. Sollte die CAS-Software defekt sein, trägt der Anwender alle Kosten für Reparatur oder Fehlerbeseitigung.

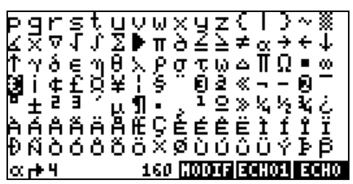
Wenn nicht durch geltendes Recht ausgeschlossen, übernimmt der Copyright-Besitzer keinerlei Haftung für Schäden, einschließlich allgemeiner, spezieller, direkter oder indirekter Schäden, die sich aus der Verwendung der CAS-Software ergeben (einschließlich aber nicht beschränkt auf Datenverlust, falscher Datenverarbeitung oder Verlusten, die der Anwender oder Dritte durch Fehler der CAS-Software in der Zusammenarbeit mit anderen Programmen erleiden), selbst wenn der Anwender oder Dritte auf die Möglichkeit von Schäden hingewiesen wurden. Wenn durch geltendes Recht erforderlich, kann der Schadensersatz, der vom Copyright-Besitzer zu zahlen ist, den Betrag, den Hewlett-Packard dem Copyright-Besitzer für die CAS-Software gezahlt hat, nicht überschreiten.

Anhang D
Zusätzlicher Zeichensatz

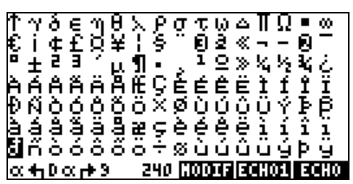
Sie können jeden der englische Gross- und Kleinbuchstaben sowie alle Ziffern direkt über die Tastatur erreichen, doch Sie können insgesamt 255 verschiedene Zeichen mit dem Rechner benutzen   einschliesslich Sonderzeichen wie θ, λ, etc., die in algebraischen Ausdrücken verwendet werden. Um auf diese Zeichen zuzugreifen, benutzen wir die Tastenkombination ↱ CHARS (zusammen mit der EVAL Taste). Auf dem Display wird folgendes angezeigt:



Mit den Pfeitasten, ◀▶▼▲, können wir in diesem Zeichensatz navigieren. Die Bewegung nach unten bewirkt zum Beispiel die Anzeige weiterer Zeichen auf dem Display:



Weiter unten sehen wir folgende Zeichen:



Ein Zeichen ist ständig hervorgehoben. Die untere Zeile des Displays zeigt das Tastenkürzel des hervorgehobenen Zeichens, sowie den ASCII Code des

Zeichens. (z.B. ist in der obigen Anzeige das Tastenkürzel α←Dα→9, also ALPHA ← D ALPHA → 9, und der ASCII Code ist *240*). Das Display zeigt auch drei Funktionen, die den Softmenü Tasten D, E, und F zugeordnet sind. Diese Funktionen sind:

MODIF: Öffnet eine grafische Anzeige mit der der Benutzer das hervorgehobene Zeichen verändern kann Seien Sie vorsichtig mit dieser Funktion, denn sie ändert das Zeichen bis zum nächsten Reset des Rechners. (Stellen Sie sich den Effekt vor, wenn Sie die Anzeige des Zeichens 1 so abändern dass es aussieht wie eine 2!).

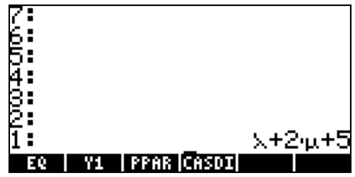
ECHO1: Kopiert das hervorgehobene Zeichen in die Befehlszeile oder den Equation Writer (EQW) und verlässt die Zeichensatz-Anzeige (d.h. gibt ein einzelnes Zeichen in den Stack zurück).

ECHO: Kopiert das hervorgehobene Zeichen in die Befehlszeile oder den Equation Writer (EQW), doch der Cursor verbleibt in der Zeichensatz-Anzeige, damit der Benutzer weitere Zeichen auswählen kann (d.h., gibt eine Zeichenfolge in den Stack zurück). Um die Zeichensatz-Anzeige zu verlassen, drücken Sie `.

Nehmen wir zum Beispiel an, Sie müssen den Ausdruck $\lambda^2 + 2\mu + 5$ eingeben.

Hier ein Vorschlag zur Vorgehensweise, wobei wir den Stack entweder im ALG Modus oder im RPN Modus benutzen:

Mit der Tastenkombination ' → CHARS kommen Sie in die Zeichensatz-Anzeige. Als nächstes benutzen Sie die Pfeiltasten, um das Zeichen λ.hervorzuheben. Drücken Sie ECHO1 (d.h. die F5 Taste), und fahren Sie fort mit der Tastenkombination + 2 × → CHARS. Nun benutzen Sie die Pfeiltasten, um das Zeichen μ.hervorzuheben Drücken Sie ECHO1 (d.h. die F5 Taste), und beenden Sie den Ausdruck mit der Tastenkombination + 5 ENTER. Hier sind die Resultate dieser Übung sowohl im ALG- als auch im RPN Modus:



Im folgenden listen wir die gebräuchlichsten [ALPHA][↱] Tastenkombinationen auf:

Griechische Buchstaben

α	(alpha)	[ALPHA][↱][A]
β	(beta)	[ALPHA][↱][B]
δ	(delta)	[ALPHA][↱][D]
ε	(epsilon)	[ALPHA][↱][E]
θ	(theta)	[ALPHA][↱][T]
λ	(lambda)	[ALPHA][↱][N]
μ	(mu)	[ALPHA][↱][M]
ρ	(rho)	[ALPHA][↱][F]
σ	(sigma)	[ALPHA][↱][S]
τ	(tau)	[ALPHA][↱][U]
ω	(omega)	[ALPHA][↱][V]
Δ	(upper-case delta)	[ALPHA][↱][C]
Π	(upper-case pi)	[ALPHA][↱][P]

Andere Zeichen

~	(tilde)	[ALPHA][↱][1]
!	(factorial)	[ALPHA][↱][2]
?	(question mark))	[ALPHA][↱][3]
\	(backward slash)	[ALPHA][↱][5]
∡	(angle symbol)	[ALPHA][↱][6]
@	(at)	[ALPHA][↱][ENTER]

Einige gebräuchliche Zeichen, die nicht über einfache Tastenkombinationen erreicht werden können sind: $\bar{x}$ (x bar), γ (gamma), η (eta), Ω (Grossbuchstabe omega). Diese Zeichen müssen aus der Zeichensatz-Anzeige CHARS "geechot" werden: ↱ CHARS .

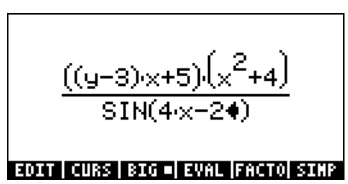
Anhang E
Auswahlbaum im EquationWriter

Der Ausdruck Baum ist ein Diagramm, das anzeigt, wie der EquationWriter einen Ausdruck darstellt (interpretiert). Die Form des Ausdrucksbaums hängt von gewissen Regeln ab, bekannt als Operations-Hierarchie. Die Regeln lauten wie folgt:

1. Operationen in Klammern werden zuerst ausgeführt, beginnend von der innersten zur äußersten Klammer und von links nach rechts innerhalb des Ausdrucks.
2. Argumente von Funktionen, werden als nächstes, von links nach rechts ausgeführt.
3. Anschließend folgen die Funktionen, ebenfalls von links nach rechts.
4. Zahlenpotenzen, werden danach von links nach rechts ausgeführt.
5. Multiplikationen und Divisionen folgen dem, ebenfalls von links nach rechts.
6. Additionen und Subtraktionen werden als letzter Rechenvorgang von links nach rechts ausgeführt.

Ausführung von links nach rechts bedeutet, wenn zwei Operationen derselben Hierarchie, sagen wir zwei Multiplikationen, in einem Ausdruck vorhanden sind, wird als erstes die links befindliche Multiplikation vor der zweiten, rechts von dieser befindlichen, usw. ausgeführt.

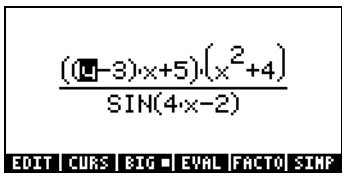
Nehmen wir als Beispiel den nachfolgenden Ausdruck, angezeigt im EquationWriter:



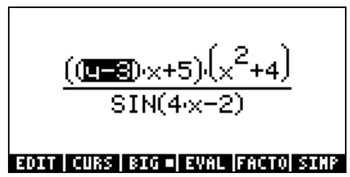
Der Einfügecursor (⬅) an dieser Stelle befindet sich rechts der 2 im Argument der SIN Funktion im Nenner. Drücken Sie die Pfeiltaste ▼, um den reinen

Bearbeitungscursor (□) um die 2 herum, im Nenner zu erhalten. Als nächstes drücken Sie die linke Pfeiltaste so lange, bis sich der reine Bearbeitungscursor um das y herum im ersten Faktor des Nenners befindet. Drücken Sie anschließend die Pfeiltaste nach oben, um den Auswahlcursor (■) um das y herum zu erhalten. Drücken Sie nun wiederholt die Pfeiltaste nach oben , wir können dem Ausdrucksbaum folgen, welcher nun bei y beginnt und bis hin zur Komplettierung des Ausdrucks folgt. Nachfolgend ist die Sequenz der Operationen, welche von der Pfeiltaste nach oben hervorgehoben wurde:

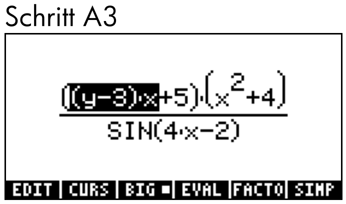
Schritt A1



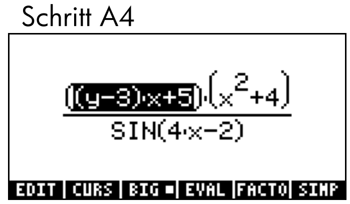
Schritt A2



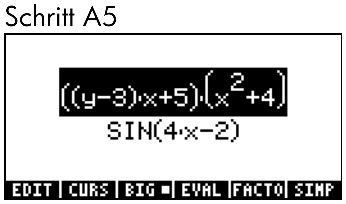
Schritt A3



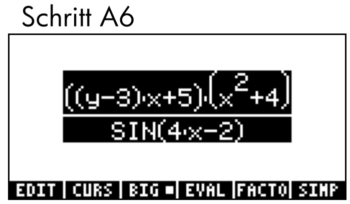
Schritt A4



Schritt A5

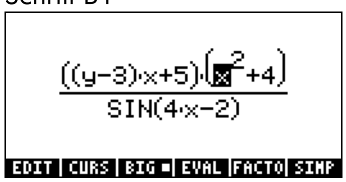


Schritt A6

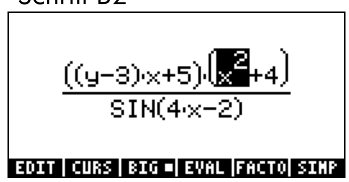


Beachten wir die Anwendung der Operations-Hierarchie in dieser Auswahl. Als erstes das y (Schritt A1). Dann, y-3 (Schritt A2, Klammern). Dann (y-3)x (Schritt A3, Multiplikation). Dann (y-3)x+5, (Schritt A4, Addition). Dann, ((y-3)x+5)(x^2+4) (Schritt A5, Multiplikation) und schließlich ((y-3)x+5)(x^2+4)/SIN(4x-2) (Schritt A6, Division). Es ist wichtig hervorzuheben, dass die Multiplikation in Schritt A5 das erste Glied ((y-3)x+5) mit einem zweiten Glied (x^2+4), welches bereits berechnet wurde, einschließt. Um die Schritte in der Kalkulation des zweiten Gliedes zu sehen, drücken Sie wiederholt die Taste Pfeil nach unten bis sich der reine Bearbeitungscursor um das y herum befindet, ein weiteres Mal. Drücken Sie anschließend die Pfeiltaste bis sich dieser Cursor über dem x im zweiten Glied des Nenners befindet. Drücken Sie dann die Taste Pfeil nach oben, um dieses x auszuwählen. Die Schritte in der Berechnung dieses Ausdrucks, wenn wir von dieser Stelle aus starten, wird wie nachfolgend gezeigt aussehen:

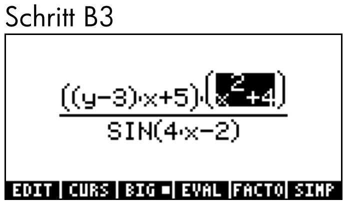
Schritt B1



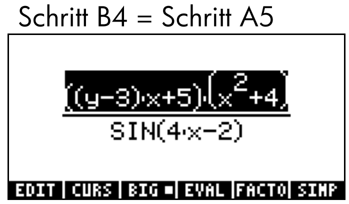
Schritt B2



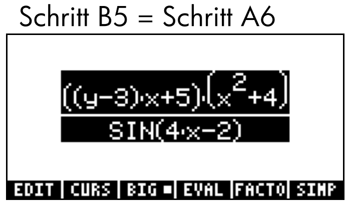
Schritt B3



Schritt B4 = Schritt A5

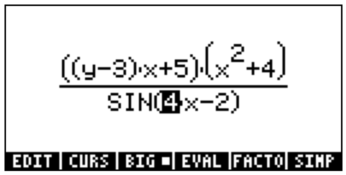


Schritt B5 = Schritt A6

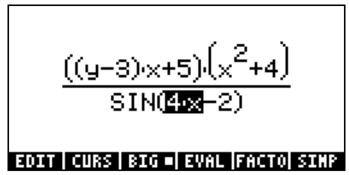


Wir können die Berechnung des Ausdrucks auch beginnend mit der 4 im Argument der SIN Funktion im Nenner starten. Drücken Sie wiederholt die Taste Pfeil nach unten , bis sich der reine Bearbeitungscursor über dem y, ein weiteres Mal befindet. Drücken Sie dann die rechte Pfeiltaste, bis sich der Cursor über der 4 im Nenner befindet. Drücken Sie dann die Taste Pfeil nach oben ▲, um diese 4 auszuwählen. Die Schritte in der Berechnung dieses Ausdrucks, wenn wir von dieser Stelle aus starten, wird nachfolgend gezeigt:

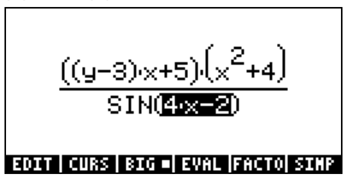
Schritt C1



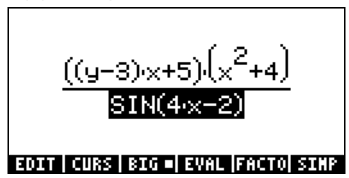
Schritt C2



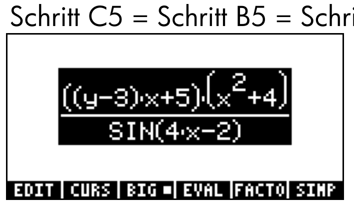
Schritt C3



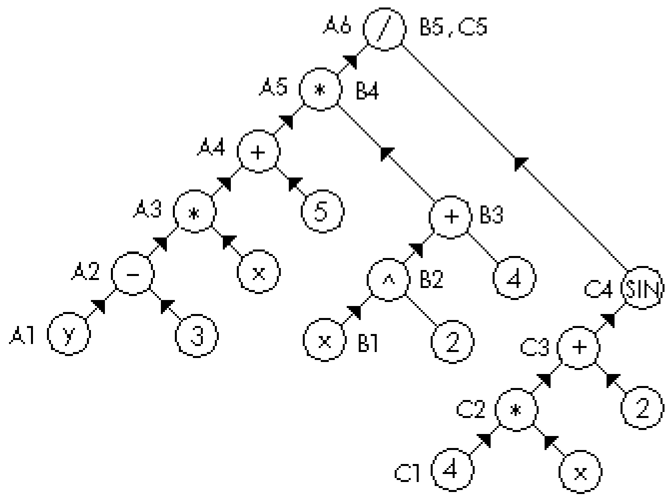
Schritt C4



Schritt C5 = Schritt B5 = Schritt A6



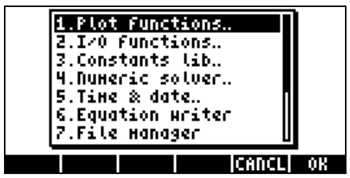
Der Ausdrucksbaum für den oben dargestellten Ausdruck wir nachfolgend gezeigt:

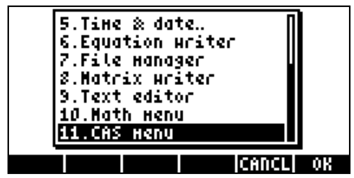


Die Schritte bei der Berechnung der Glieder des Baumes (A1 bis A6, B1 bis B5 und C1 bis C5) werden neben dem Kreis, welcher Zahlen, Variablen oder Operatoren enthält, angezeigt.

Anhang F
Das Applications (APPS) Menü

Das Applications (APPS) Menü ist erreichbar mit der APPS Taste (erste Taste in der zweiten Reihe von oben). Die APPS Taste zeigt die folgenden Anwendungen:

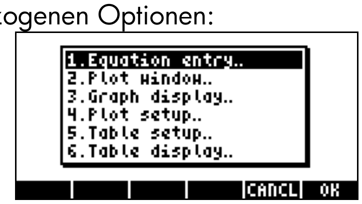




Die verschiedenen Anwendungen werden im folgenden beschrieben.

Plot Funktionen

Die Auswahl von Option *1. Plot functions..* im APPS Menü zeigt die folgende Menüliste mit grafikbezogenen Optionen:



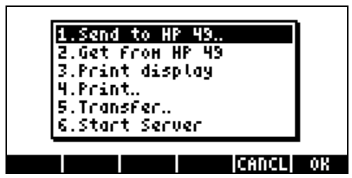
Die sechs gezeigten Optionen entsprechen den unten aufgeführten Tastenkombinationen:

Equation entry…	← Y=	Plot window..	← WIN
Graph display..	← GRAPH	Plot setup..	← 2D/3D
Table setup..	← TBLSET	Table display..	← TABLE

Diese Applikationen werden in Kapitel 12 detailliert vorgestellt

I/O functions..

Die Auswahl von Option *2. I/O functions..* im APPS Menü zeigt die folgende Menüliste von input/output Funktionen

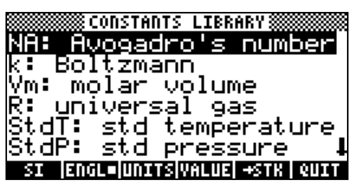


Diese Anwendungen werden als nächstes beschrieben:

Send to HP 49..	Daten an einen anderen Rechner senden
Get from HP 49	Daten von einem anderen Rechner empfangen
Print display	Bildschirm ausdrucken
Print..	ausgewähltes Objekt drucken
Transfer..	Daten an ein anderes Gerät schicken
Start Server..	Den Rechner als Server zur Kommunikation mit Computern einrichten

Constants lib..

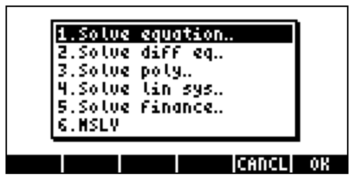
Die Auswahl von Option *3. Constants lib..* im APPS Menü öffnet die Constant Library Application, die Werte für physikalische Standard Konstanten liefert:



Die Konstanten-Bibliothek wird detaillliert in Kapitel 3 besprochen.

Numeric solver..

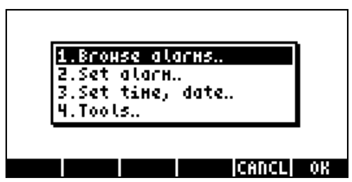
Die Auswahl von Option *3. Constants lib..* im APPS Menü liefert das Numerical Solver Menu:



Diese Operation entspricht der Tastenkombination NUM.SLV . Das Numerical Solver Menü wird detailliert in den Kapiteln 6 und 7 vorgestellt.

Time & date..

Bei Auswahl von Option *5.Time & date..* im the APPS Menü erscheint das Zeit und Datum Menü:



Diese Operation entspricht der Tastenkombination TIME . Das Zeit und Datum Menü wird im Detail in Kapitel 26 vorgestellt.

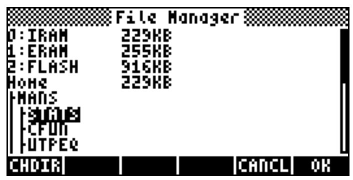
Equation writer..

Die Auswahl von Option *6.Equation writer..* im APPS Menu öffnet den equation writer:

Diese Operation entspricht der Tastenkombination ⓡ EQW . Der equation writer wird detailliert in Kapitel 2 besprochen. Beispiele für den Gebrauch des equation writers finden Sie in diesem Handbuch.

File manager..

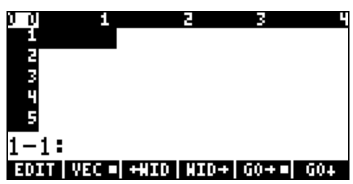
Option *7.File manager..* des APPS Menüs startet den file manager:



Diese Operation entspricht der Tastenkombination ⓛ FILES .Der file manager wird in Kapitel 2 erklärt.

Matrix writer..

Option *8.Matrix writer..* des APPS Menüs startet den matrix writer:



Diese Operation entspricht der Tastenkombination ⓛ MTRW .Der matrix writer wird detailliert in Kapitel 10 erklärt

Text editor..

Option *9.Text editor..* des APPS startet den Zeilen Text Editor:

Der Text Editor kann in vielen Fällen durch drücken der ▼ Taste gestartet werden   Wenn das im Display angezeigte Objekt ein algebraisches Objekt ist, wird mit der ▼ Taste in der Regel der Equation Writer gestartet. Der text editor wird in Kapitel 2 vorgestellt, und im Detail in Appendix L erklärt

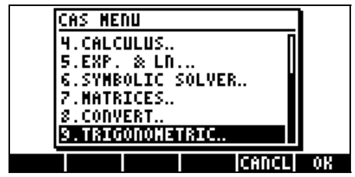
Math Menü

Die Option *10.Math menu..* im APPS öffnet das MTH (mathematics) Menü:

Diese Operation entspricht der Tastenkombination ← MTH . Das MTH Menu wird vorgestellt in Kapitel 3 (reale Zahlen) und in den Kapiteln 4 (komplexe Zahlen), 8 (Listen), 9 (Vectoren), 10 (Erzeugen von Matrixen), 11 (Matrix Operationen), 16 (schnelle Fourier Transformationen), 17 (Wahrscheinlichkeitsberechnungen), und 19 (Zahlensysteme).

CAS menu..

Option *11.CAS menu..* des APPS Menüs öffnet das CAS oder SYMBOLIC Menü:



Das Menü erscheint auch durch drücken der Taste SYMB. Das CAS oder SYMBOLIC Menü wird vorgestellt in den Kapiteln 5 (algebraische und arithmetische Operationen), 4 (komplexe Zahlen), 6 (Lösung von Gleichungen), 10 (Erzeugen von Matrixen), 11 (Matrix Operationen), 13 (Kalkulus), 14 (Kalkulus mit mehreren Variablen), und 15 (Vektor Analyse).

Anhang G
Nützliche Tastenkürzel

Hier zeigen wir Ihnen eine Reihe von Tastaturkürzeln die üblicherweise mit diesem Rechner benutzt werden:

- Einstellen des Display-Kontrasts: ON (festhalten) +, oder ON (festhalten) −

- Umschalten zwischen RPN und ALG Modus: MODE +/- OK

- System Flag 95 setzen bzw. entfernen (ALG bzw.. RPN Arbeits Modus)

  MODE FLAGS ▲ ↰ ▲ ↰ ▲ ↰ ▲ CHK

  - Im ALG Modus
    CF(-95) wählt den RPN Modus

  - Im RPN Modus
    95 +/- ENTER SF wählt den ALG Modus

  Ein Tastaturkürzel zum Umschalten zwischen APPROX und EXACT Modus ist das folgende: ↱ (festhalten) ENTER.

- Setzen/Entfernen des System Flags 105 (EXACT bzw. APPROX CAS Modus)

  MODE FLAGS ▲ ↰ ▲ ↰ ▲ ▲ ▲ CHK

  - Im ALG Modus,
    SF(-105) wählt den APPROX CAS Modus
    CF(-105) wählt den EXACT CAS Modus

  - Im RPN Modus,
    105 +/- ENTER SF) wählt den APPROX CAS Modus
    105 +/- ENTER CF) wählt den EXACT CAS Modus

- Setzen/Entfernen des System Flags 117 (CHOOSE BOXES bzw. SOFT Menüs):

  MODE FLAGS ▲ ↰ ▲ ▼ CHK

  - In ALG Modus,
    SF(-117) wählt die SOFT Menüs
    CF(-117) wählt die CHOOSE BOXES.

  - In RPN Modus,
    117 +/- ENTER SF wählt die SOFT Menüs
    117 +/- ENTER CF wählt die SOFT Menüs

- Winkelmassänderung:
  - zu Grad: ALPHA ALPHA D E G ENTER
  - zu Radiant: ALPHA ALPHA R A D ENTER

- Sonderzeichen:
  - Winkel Symbol (∠): ALPHA ↱ 6
  - Factorprodukt Symbol (!): ALPHA ↱ 2
  - Grad Symbol (°): ALPHA ↱ (festhalten) 6

- Sperren/Entsperren der Alpha Tastatur:
  - Sperren der Alpha Tastatur (Grossbuchstaben): ALPHA ALPHA
  - Entsperren der Alpha Tastatur (Grossbuchstaben): ALPHA
  - Sperren der Alpha Tastatur (Kleinbuchstaben):
    ALPHA ALPHA ↰ ALPHA
  - Entsperren der Alpha Tastatur (Kleinbuchstaben):
    ↰ ALPHA ALPHA

- Griechische Buchstaben:

Alpha (α):	ALPHA ↱ A	Beta (β):	ALPHA ↱ B
DELTA (Δ):	ALPHA ↱ C	Delta (d):	ALPHA ↱ D
Epsilon (ε):	ALPHA ↱ E	Rho (ρ):	ALPHA ↱ F
Mu (μ):	ALPHA ↱ M	Lambda (λ):	ALPHA ↱ N
PI (Π):	ALPHA ↱ P	Sigma (σ):	ALPHA ↱ S

Theta (θ): ALPHA ↱ T   Tau (t): ALPHA ↱ U
Omega (ω): ALPHA ↱ V

- System-Level Betrieb (Halten Sie ON gedrückt, lassen Sie die Taste los nachdem Sie eine zweite oder dritte Taste gedrückt haben):
    - ON (festhalten) F1 F6 : "Kaltstart" – aller Speicher wird gelöscht
    - ON (festhalten) F2 : Cancelt Tastenbetätigung
    - ON (festhalten) F3 : "Warmstart" – Speicherinhalt bleibt erhalten
    - ON (festhalten) F4 : Startet interaktiven Selbsttest
    - ON (festhalten) F5 : Startet kontinuierlichen Selbsttest
    - ON (festhalten) SPC : Deep-Sleep Shutdown - Timer aus
    - ON (festhalten) F1 : Erzeugt einen "Screendump"
    - ON (festhalten) F4 : Cancelt den nächsten zu wiederholenden Alarm

- Menüs auf die nicht über die Tastatur zugegriffen werden kann: Im RPN Modus geben Sie die "MENÜ" (Menu_number) ein. Im ALG Modus, geben Sie "MENÜ" (Menu_number) ein. Menu_number ist einer der folgenden Werte:
    - STAT Soft Menü: 96
    - PLOT Soft Menü: 81
    - SOLVE Soft Menü: 74, oder ↱ (festhalten) 7
    - UTILITY Soft Menü: 113

- Andere Menüs:
    - MATHS Menü: ALPHA ALPHA M A T H S ENTER
    - MAIN Menü: ALPHA ALPHA M A I N ENTER

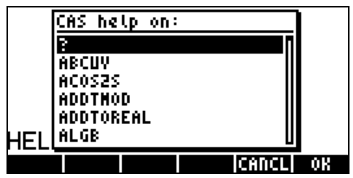
- Andere Tastaturkürzel:
    - ↱ (festhalten) 7 : SOLVE Menü (Menü 74)
    - ↰ (festhalten) MODE : PRG/MODUS Menü (Kapitel 21)
    - ↰ (festhalten) ▼ : Startet den Text Editor (Appendix L)
    - ↰ (festhalten) UPDIR : HOME(), springt ins HOME

Verzeichnis

- ◯ [←] (festhalten) PREV : Wiederherstellen des letzten aktiven Menüs
- ◯ [→] (festhalten) ▼ : Listet den Inhalt von Variablen oder Menü Einträge
- ◯ [→] (festhalten) CHARS : PRG/CHAR Menü (Kapitel 21)

Anhang H
CAS Hilfe System

Listen des CAS-Hilfes Systems sie können mit der Tastenkombination TOOL NXT HELP ENTER darauf zugreifen. Die ersten Hilfe-Bildschirme sind nachstehend abgebildet:



Die Befehle sind in alphabetischer Folge aufgelistet. Mit Hilfe der Pfeiltasten ▲▼ können Sie sich durch die Hilfe-Liste bewegen. Im Folgenden finden Sie einige nützliche Hinweise zur Steuerung des Hilfesystems.

- Sie können die Pfeil-unten-Taste ▼ gedrückt halten und so lange auf den Bildschirm schauen, bis der gesuchte Befehl erscheint. In diesem Moment lassen Sie die Pfeil-unten-Taste los. Aller Wahrscheinlichkeit nach wird der gesuchte Befehl in diesem Moment nicht ausgewählt sein (Sie werden eventuell über das Ziel hinausschießen oder davor stoppen). Allerdings können Sie die Vertikaltasten ▲▼ einzeln drücken, um den gewünschten Befehl auszuwählen und dann OK drücken.
- Falls Sie bei niedergehaltener Pfeil-unten-Taste ▼ über den gesuchten Befehl hinausschießen, können Sie die Pfeil-oben-Taste ▲ gedrückt halten – um sich wieder rückwärts zum gesuchten Befehl hin zu bewegen. Verfeinern Sie Ihre Auswahl durch Einzelbetätigung der Vertikaltasten ▲▼.
- Sie können auch den Anfangsbuchstaben des gesuchten Befehls tippen und dann die Pfeil-unten-Taste ▼ zum Auffinden des jeweiligen Befehls verwenden. Angenommen, Sie suchen nach dem Befehl DERIV: Nach dem Aufrufen des Hilfesystems (TOOL NXT HELP

ENTER) tippen Sie ALPHA D . Dadurch wird der erste Befehl ausgewählt, der mit D beginnt, z. B. DEGREE. Um DERIV zu finden, drücken Sie zweimal ▼. Um den Befehl auszuwählen, drücken Sie OK.

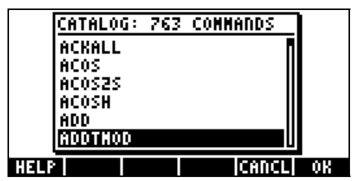
- Durch Sperren der alphabetischen Tastatur können Sie zwei oder mehr Anfangsbuchstaben des gesuchten Befehls eingeben. Dadurch werden Sie direkt zum gesuchten Befehl oder zumindest in seine Nähe gebracht. Danach müssen Sie die Feststellung der alphabetischen Tastatur wieder lösen und die vertikalen Pfeiltasten ▲▼ zum Auffinden des Befehls verwenden, sofern nötig. Drücken Sie OK, um den Befehl auszuwählen. Um beispielsweise den Befehl PROPFRAC zu finden, können Sie eine der nachstehenden Tastenfolgen verwenden:

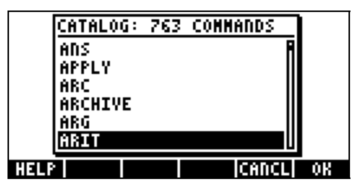
TOOL NXT HELP ENTER ALPHA ALPHA P R ALPHA ▼ ▼ OK
TOOL NXT HELP ENTER ALPHA ALPHA P R O ALPHA ▼ OK
TOOL NXT HELP ENTER ALPHA ALPHA P R O P ALPHA OK

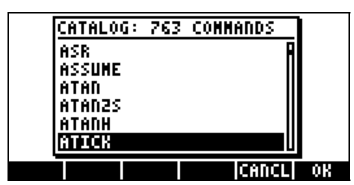
In Anhang C finden Sie weitere Informationen zu CAS (Computer Algebraic System). Anhang C enthält auch weitere Anwendungsbeispiele des CAS-Hilfesystems.

Anhang I
Befehlsliste

Dies ist eine Liste aller zur Verfügung stehenden Befehle des Befehlskatalogs (CAT). Die Bedfehle, die zum CAS (Computer Algebraic System) gehören, sind auch in Appendix H aufgeführt. CAS Hilfesystem-Einträge sind für einen Befehl verfügbar, wenn die Softmenü Taste HELP beim Hervorheben des gewünschten Befehls erscheint. Drücken Sie diese Softmenü Taste um die CAS Hilfe für den Befehl zu erhalten. Die ersten Hilfsbildschirme des Katalogs sehen Sie unten:



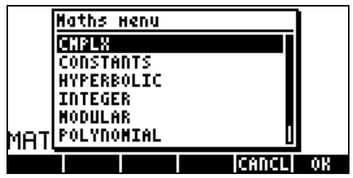




Die vom Anwender definierten Befehle erscheinen auch in der Liste des Befehle Katalogs, in Kursivschrift. Ist eine Hilfefunktion zur Bibliothek vorhanden, erscheint die Funktionstaste HELP , sobald Sie einen anwenderdefinierten Befehl, anklicken für den ein Hilfetext hinterlegt wurde.

Anhang J
Das MATHS Menü

Das MATHS Menü, auf das mit dem Befehl MATHS (verfügbar im Befehlskatalog _CAT) zugegriffen werden kann, enthält folgende Untermenüs;



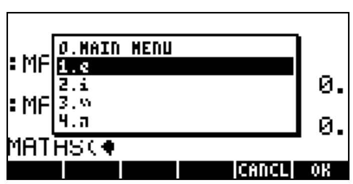
Das CMPLX Untermenü

Das CMPLX Untermenü enthält Funktionen, die für Operationen mit komplexen Zahlen hilfreich sind:

Diese Funktionen sind in Kapitel 4 beschrieben 4.

Das CONSTANTS Untermenü

Das CONSTANTS Untermenü erlaubt den Zugriff auf die im Rechner verfügbaren mathematischen Konstanten. Diese sind beschrieben in Kapitel 3:

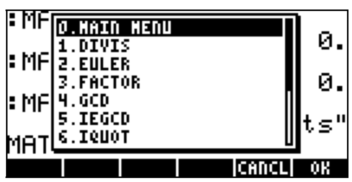


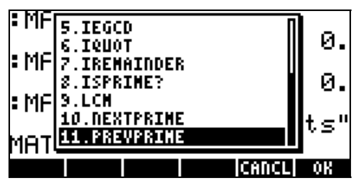
Das HYPERBOLIC Untermenü

Das HYPERBOLIC Untermenü enthält die Hyperbel Funktionens und ihre Umkehrfunktionen. Diese sind beschrieben in Kapitel 3:

Das INTEGER Untermenü

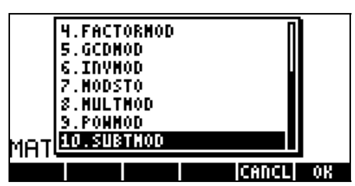
Das INTEGER Untermenü stellt Funktionen für die Manipulation ganzer Zahlen sowie einiger Polynome zur Verfügung. Diese Funktionen werden beschrieben in Kapitel 5:





Das MODULAR Untermenü

Das MODULAR Untermenü stellt Funktionen für modulare Arithmetik mit Zahlen und Polynomen zur Verfügung. Diese Funktionen werden beschrieben in Kapitel 5:

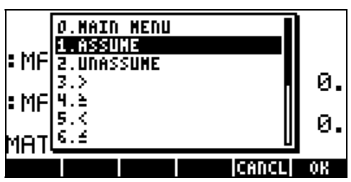


Das POLYNOMIAL Untermenü

Das POLYNOMIAL Untermenü beinhaltet Funktionen zur Erzeugung und Manipulation von Polynomen. Diese Funktionen werden beschrieben in Kapitel 5:

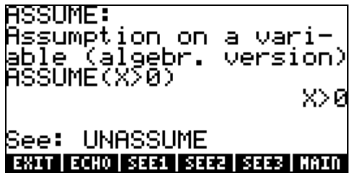
Die TESTS Untermenü

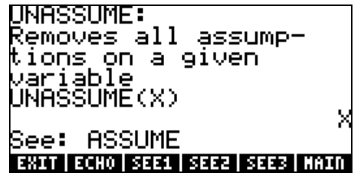
The TESTS Untermenü beinhaltet relationale Operatoren (==, <, etc.), logische Operatoren (AND, OR, etc.), die IFTE Funktion, und die Befehle ASSUME und UNASSUME.



Relationale und logische Operatoren werden im Kapitel 21 im Zusammenhang mit der Programmierung des Rechners in der Programmiersprache RPL vorgestellt. Die IFTE Funktion wird in Kapitel 3 eingeführt. Die Funktionen ASSUME und UNASSUME werden als nächstes präsentiert, indem wir deren CAS Hilfesystem Einträge benutzen (siehe Appendix C).

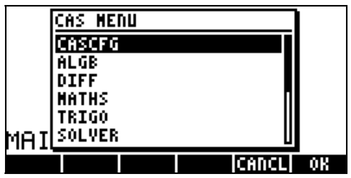
ASSUME **UNASSUME**


ASSUME:
Assumption on a vari-
able (algebr. version)
ASSUME(X>0)
X>0
See: UNASSUME
EXIT ECHO SEE1 SEE2 SEE3 MAIN


UNASSUME:
Removes all assump-
tions on a given
variable
UNASSUME(X)
X
See: ASSUME
EXIT ECHO SEE1 SEE2 SEE3 MAIN

Anhang K
Das MAIN Menü

Das MAIN Menü ist verfügbar im Befehlskatalog Es enthält folgende Untermenüs:



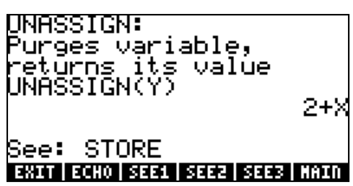
Der CASCFG Befehl

Dies ist der erste Eintrag im MAIN Menü. Dieser Befehl konfiguriert das CAS. Für Informationen über die CAS Konfiguration siehe Anhang C.

Das ALGB Untermenü

Das ALGB Untermenü enthält die folgenden Befehle:

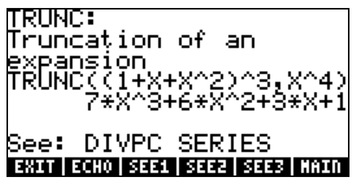
Diese Funktionen, mit Ausnahme von 0.MAIN MENÜ and 11.UNASSIGN sind verfügbar im ALG Tastatur Menü (⮣ ALG). Detaillierte Erklärungen dieser Funktionen finden Sie in Kapitel 5.   Die Funktion UNASSIGN ist in folgendem Eintrag des CAS Menüs beschrieben:



Das DIFF Untermenü

Das DIFF Untermenü enthält die folgenden Funktionen:

Diese Functionen sind auch verfügbar im CALC/DIFF Untermenü (⬅ CALC). Sie sind beschrieben in den Kapiteln 13, 14, and 15, mit Ausnahme der Funktion TRUNC, die als nächstes anhand ihres CAS Hilfesystem-Eintrags beschrieben wird.



Das MATHS Untermenü

Das MATHS Menü wird detalliert in Anhang J beschrieben.

Das TRIGO Untermenü

Das TRIGO Menü enthält die folgenden Funktionen:

Diese Funktionen sind auch verfügbar im TRIG Menü (TRIG). Die Beschreibung dieser Funktionen erfolgt in Kapitel 5.

Das SOLVER Untermenü

Das SOLVER Menü enthält die folgenden Funktionen:

Diese Funktionen sind verfügar im CALC/SOLVE Menü (CALC). Sie werden beschrieben in den Kapiteln 6, 11, and 16.

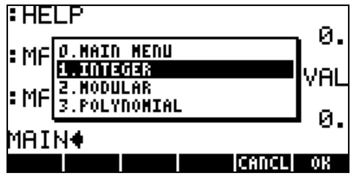
Das CMPLX Untermenü

Das CMPLX Menü enthält die folgenden Funktionen:

Das CMPLX Menü ist auch über die Tastatur erreichbar (CMPLX). Einige dieser Funktionen des CMPLX Menüs sind auch im MTH/COMPLEX Menü verfügbar (MTH). Funktionen für komplexe Zahlen werden in Kapitek 4 vorgestellt.

Das ARIT Untermenü

Das ARIT Menü enthält die folgenden Untermenüs:



Die Untermenüs INTEGER, MODULAR, und POLYNOMIAL werden detailliert in Anhang J vorgestellt.

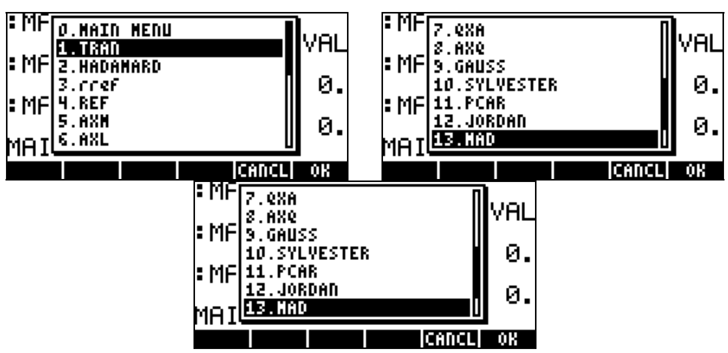
Das EXP&LN Untermenü

Das EXP&LN Menü enthält die folgenden Funktionen:

Auf das Menü kann auch über die Tastatur zugegriffen werden mit EXP&LN . Die Funktionen dieses Menüs werden in Kapitel 5 vorgestellt.

Das MATR Submenü

Das MATR Menü enthält die folgenden Funktionen:



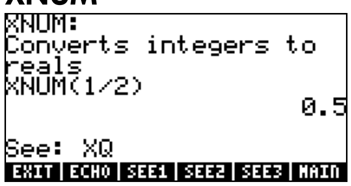
Diese Funktionen sind ebenfalls verfügbar im MATRICES Menü der Tastatur (MATRICES). Sie werden in den Kapiteln 10 und 11 beschrieben.

Das REWRITE Untermenü

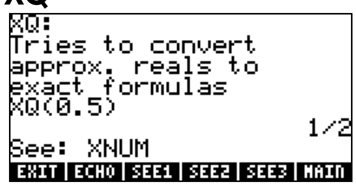
Das REWRITE Menü enthält die folgenden Funktionen:

Diese Funktionen sind verfügbar im CONVERT/REWRITE Menü (← CONVERT). Sie werden in Kapitel 5 vorgestellt mit Ausnahme der Funktionen XNUM und XQ, die als nächstes anhand ihrer korrespondierenden Einträge im CAS Hilfsystem besprochen werden. (TOOL NXT HELP):

XNUM

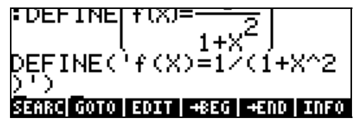


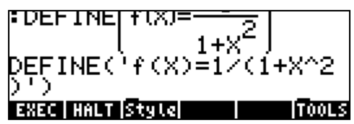
XQ



Anhang L
Zeileneditor Befehle

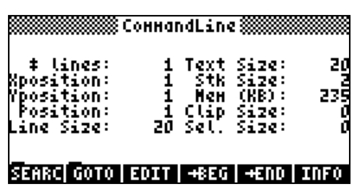
Wenn Sie mit ↰ ▼ den Zeileneditor im RPN Stack oder im ALG Modus aufrufen, werden Ihnen die folgenden Untermenü zur Verfügung gestellt (drücken Sie NXT um die restlichen Funktionen zu sehen):





Die Funktionen werden wie folgt kurz beschrieben:

←SKIP: überspringt Zeichen am Anfang eines Wortes
SKIP→: überspringt Zeichen zum Ende des Wortes
←DEL: löschen eines Zeichens am Wortanfang
DEL→: löschen von Zeichen am Ende eines Wortes
DEL L: löschen von Zeichen in der Zeile
INS: wenn ausgewählt, werden Zeichen an der Cursor-Position eingefügt wenn nicht ausgewählt, werden Zeichen durch den Cursor ersetzt (überschrieben)
EDIT: editert den ausgewählten Bereich
→BEG: bewegt den Cursor zum Wortanfang
→END: markiert das Ende eines ausgeählten Bereichs
INFO: gibt Information zum Befehlszeileneditor, z.B.:,



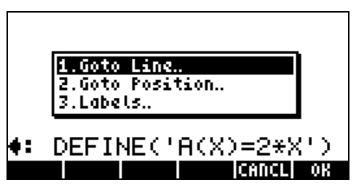
Dieser Bildschirm ist selbsterklärend. Zum Beispiel, *X* and Y *positions* bezeichnen die Position auf einer Zeile (X) und die Zeilennummer (Y). *Stk Size* bezeichnet die Nummer von Objekten in der ALG Modus History oder im RPN Stack. *Mem(KB)* bezeichnet die Grösse des freien Speichers. *Clip Size* ist die Anzahl der Zeichen im Clipboard. *Sel Size* ist die Anzahl der Zeichen im ausgewählten Bereich.

EXEC: führt den gewählten befehl aus
HALT: stoppt die Befehlsausführung.

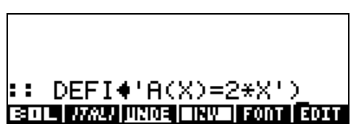
Der Zeileneditor stellt die folgenden Untermenüs zur Verfügung:

SEARCH: sucht Zeichen oder Worte in der Befehlszeile. Es umfasst folgende Funktionen:

GOTO: springt zu einer gewünschten Position in der Befehlszeile. Es enthält folgende Funktionen:



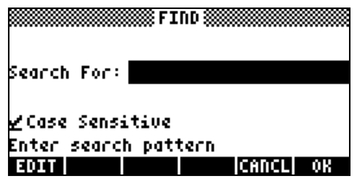
Style: Text-Auszeichnungen, die in der Befehlszeile verwendet werden können:



Das SEARCH Untermenu

Die Funktionen des SEARCH Untermenüs sind:

Find : Benutzen Sie diese Funktion, um einen String in der Befehlszeile zu finden. Dies ist das Eingabeformular, dass diesen Befehl zur Verfügung stellt:



Replace: Benutzen Sie diesen Befehl, um einen String zu finden und zu ersetzen Dies ist das Eingabeformular für diesen Befehl:

Find next..: Findet das nächste Muster das den im Find-Befehl definierten Kriterien entspricht

Replace Selection: Ersetzt den ausgewählten Bereich mit dem Ersatz Muster, das mit dem Replace Befehl definiert worden ist.

Replace/Find Next: Ersetzt ein Muster und sucht nach der nächsten Entsprechung des Musters wie in definiert im Replace Befehl.

Replace All: Ersetzt jedes Vorkommen eines bestimmten Musters. Dieser Befehl fragt. Nach einer Bestätigung durch den Benutzer, bevor ein Muster ersetzt wird.

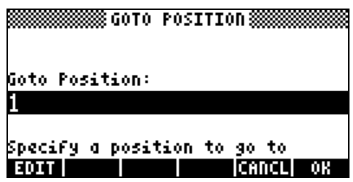
Fast Replace All: Ersetzt alle Vorkommen eines bestimmten Musters ohne Rückfrage an den Benutzer.

Das GOTO Untermenu

The functions in the GOTO sub-menu are the following:

Goto Line: springt zu einer angegebenen Zeile. Das Eingabeformular für diesen Befehl ist:

Goto Position: springt zu einer angegebenen Position in der Befehlszeile. Das Eingabeformular für diesen Befehl ist:



Labels: springt zu einem angegebenen Label in der Befehlszeile.

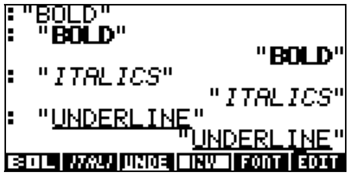
Das Style Untermenu

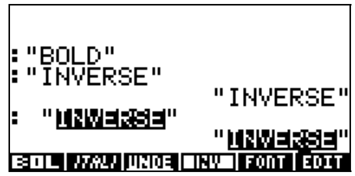
Das Style Untermenu enthält die folgendenStyles:

BOL: Bold
ITALI: Italics
UNDE: Underline
INV : Inverse

Der Befehl FONT lässt den Benutzer den Font für den Befehlszeileneditor auswählen.

Beispiele für verschiedenen Styles sehen Sie unten:


:"BOLD"
: "BOLD"
"BOLD"
: "ITALICS"
"ITALICS"
: "UNDERLINE"
"UNDERLINE"
BOL ITAL UNDE INV FONT EDIT


:"BOLD"
:"INVERSE"
"INVERSE"
: "INVERSE"
"INVERSE"
BOL ITAL UNDE INV FONT EDIT

Anhang M
Index

A

B

C

D

E

F

G

H

I

J

K

L

M

N

Q

R

S

T

U

V

W

X

Y

Z

Weitere Zeichen

Beschränkte Gewährleistung

hp 49g+ grafikrechner; Garantiezeit: 12 Monate

1. HP garantiert Ihnen, dem Endbenutzer/Kunden, dass Hardware, Zubehör und Lieferung von HP frei von Material- oder Herstellungsfehler sind. Diese Garantie erstreckt sich vom Tag des Kaufs, für die oben genannte Zeitspanne. Wird HP in der Garantiezeit über solche Schäden informiert, wird HP dieses Produkt nach eigenem Ermessen reparieren oder ersetzen. Austauschprodukte können entweder neu oder neuwertig sein.
2. HP garantiert Ihnen, dass die HP -Software, nach Kauf und für die oben genannte Zeitspanne, auch dann alle Programmierfunktionen einwandfrei ausführen wird, wenn die Material- oder Arbeitsgüte kleine Fehler aufweist, solange diese einwandfrei installiert und angewendet wird. Sollte HP während der Garantiezeit über Defekte informiert werden, wird HP die Software, die aufgrund oben erwähnter Defekten nicht einwandfrei arbeitet, ersetzen.
3. HP übernimmt aber keine Garantie für die ununterbrochene und fehlerfreie Funktion des HP Produktes. Sollte HP, innerhalb eines angemessenen Zeitraumes, nicht in der Lage sein, ein Produkt, wie gewährleistet, zu reparieren oder zu ersetzen, können Sie den Einkaufspreis umgehend zurück verlangen, wenn Sie das Gerät einschl. des Kassenbelegs zurückgeben.
4. HP Produkte können wiederaufbereitete Teile, die in ihrer Leistungsfähigkeit jedoch neuwertig sind, enthalten.
5. Diese Garantie erstreckt sich nicht auf Defekte, die aus nachfolgend aufgeführten Gründen entstanden sind (a) falsche oder unsachgemäße Wartung oder Einstellung, (b) Anwendung von Software, Schnittstellen, Teilen, die nicht von HP stammen, (c) nicht erlaubte Veränderung oder Missbrauch, (d) Anwendung außerhalb der zur Verwendung veröffentlichten Informationen oder (e) unsachgemäße Behandlung oder Wartung des Gerätes.
6. HP SCHLIESST WEITERE GARANTIEN ODER HAFTUNGEN, OB SCHRIFTLICHER ODER MÜNDLICHER NATUR, AUSDRÜCKLICH AUS. BIS ZU DEM UMFANG DEN LOKALE GESETZE ERLAUBEN, IST JEDE

GESETZLICHE GEWÄHRLEISTUNG ODER BEDINGUNG DER HANDELSÜBLICHKEIT, ZUFRIEDENSTELLENDE QUALITÄT ODER EINSATZFÄHIGKEIT FÜR EINEN BESTIMMTEN ZWECKE AUF DIE OBEN ERWÄHNTE LAUFZEIT DER GARANTIE BESCHRÄNKT. In einigen Ländern, Staaten oder Provinzen gibt es keine Einschränkungen der Laufzeit einer Garantie, Das bedeutet, dass die oben erwähnte Laufzeit oder der Haftungsausschluss auf Sie zutreffen kann oder auch nicht. Diese Garantie gibt Ihnen besondere Rechte. Sie könnten aber auch andere Rechte haben, die von Land zu Land, von Staat zu Staat oder von Provinz zu Provinz unterschiedlich sind.

7. BIS ZU DEM UMFANG, WIE ES DIE LOKALEN GESETZT ERLAUBEN, IST DER RECHTSBEHELF IN DIESER GARANTIEERKLÄRUNG IHR EINZIGER UND EXKLUSIVER RECHTSBEHELF. MIT AUSNAHME DES OBEN ERWÄHNTEN, KANN WEDER HP NOCH SEINE LIEFERANTEN, FÜR DEN VERLUST VON DATEN ODER FÜR DIREKTEN, SPEZIELLEN, ZUFÄLLIGEN, MITTELBAREN (EINSCHLIESSLICH PROFIT- ODER DATENVERLUST) VERLUST ODER SCHÄDEN, HAFTBAR GEMACHT WERDEN, SEI ES, DASS DIESE AUF DEN VERTRAG, AUF SCHADENSERSATZRECHT ODER ANDERWEITIG BASIEREN. Manche Länder, Staaten oder Provinzen erlauben keinen Ausschluss oder keine Begrenzung auf zufällige oder Folgeschäden, Daher könnte die obige Einschränkung oder der obige Ausschluss für Sie nicht zutreffen.

8. Die gewährten Garantien für HP-Produkte und –Dienstleistungen werden in den schriftlichen Garantieerklärungen aufgeführt, die diesen Produkten und Dienstleistungen beigelegt werden. HP ist nicht haftbar für technische oder redaktionelle Fehler oder Auslassungen in diesem Dokument.

FÜR ENDVERBRAUCHER TRANSAKTIONEN IN AUSTRALIEN UND NEU SEELAND: DIE IN DIESER ERKLÄRUNG ENTHALTENEN GARANTIEBEDINGUNGEN, SCHLIESSEN DIE GESETZLICHEN BESTIMMUNGEN NICHT AUS, SCHRÄNKEN DIESE NICHT EIN ODER ÄNDERN DIESE NICHT UND SIND ZUSÄTZLICH ZU DEN OBLIGATORISCHEN GESETZLICHEN RECHTEN FÜR DEN VERKAUF DES PRODUKTES ANWENDBAR.

Service

Europa

Land:	Telefonnummern
Österreich	+43-1-3602771203
Belgien	+32-2-7126219
Dänemark	+45-8-2332844
Osteuropäische Staaten	+420-5-41422523
Finnland	+35-89640009
Frankreich	+33-1-49939006
Deutschland	+49-69-95307103
Griechenland	+420-5-41422523
Holland	+31-2-06545301
Italien	+39-02-75419782
Norwegen	+47-63849309
Portugal	+351-229570200
Spanien	+34-915-642095
Schweden	+46-851992065
Schweiz	+41-1-4395358 (deutsch) +41-22-8278780 (französisch) +39-02-75419782 (italienisch)
Türkei	+420-5-41422523
Groß Britannien	+44-207-4580161
Tschechien	+420-5-41422523
Südafrika	+27-11-237 62 00
Luxemburg	+32-2-7126219
andere europäische Länder	+420-5-41422523

Asien Pazifik

Land:	Telefonnummern
Australien	+61-3-9841-5211
Singapur	+61-3-9841-5211

Lat.Amerika

Land:	Telefonnummern
Argentinien	0-810-555-5520
Brasilien	Sao Paulo 3747-7799; ROTC 0-800-157751

Mexiko	Mx City 5258-9922; ROTC 01-800-472-6684
Venezuela	0800-4746-8368
Chile	800-360999
Kolumbien	9-800-114726
Peru	0-800-10111
Mittelamerika & Karibik	1-800-711-2884
Guatemala	1-800-999-5105
Puerto Rico	1-877-232-0589
Costa Rica	0-800-011-0524

N. Amerika

Land:	Telefonnummern
USA	1800-HP INVENT
Kanada	(905) 206-4663 or 800- HP INVENT

ROTC = Rest des Landes

Regulierungsinformationen

In diesem Abschnitt finden Sie Informationen, welche zeigen, dass der hp 49g+ mit den Regelungen in bestimmten Regionen übereinstimmt. Jedwelche Änderungen, die Sie am Rechner durchführen, welche nicht ausdrücklich von Hewlett Packard genehmigt wurden, könnte Ihnen das Recht nehmen, den 49g+ in dieser Region zu benutzen.

USA

This calculator generates, uses, and can radiate radio frequency energy and may interfere with radio and television reception. The calculator complies with the limits for a Class B digital device, pursuant to Part 15 of the FCC Rules. These limits are designed to provide reasonable protection against harmful interference in a residential installation.

However, there is no guarantee that interference will not occur in a particular installation. In the unlikely event that there is interference to radio or television reception(which can be determined by turning the calculator off and on), the user is encouraged to try to correct the interference by one or more of the following measures:

- Reorient or relocate the receiving antenna.
- Relocate the calculator, with respect to the receiver.

Connections to Peripheral Devices
To maintain compliance with FCC rules and regulations, use only the cable accessories provided.

Canada
This Class B digital apparatus complies with Canadian ICES-003. Cet appareil numerique de la classe B est conforme a la norme NMB-003 du Canada.

Japan
この装置は、情報処理装置等電波障害自主規制協議会(VCCI)の基準に基づく第二情報技術装置です。この装置は、家庭環境で使用することを目的としていますが、この装置がラジオやテレビジョン受信機に近接して使用されると、受信障害を引き起こすことがあります。
取扱説明書に従って正しい取り扱いをしてください。